Lecture 5: Unscented Kalman filter, Gaussian Filter, GHKF and CKF

Simo Särkkä

Department of Biomedical Engineering and Computational Science
Aalto University

February 24, 2011
Problem: Determine the mean and covariance of $y$:

$$x \sim N(\mu, \sigma^2)$$

$$y = \sin(x)$$

**Linearization** based approximation:

$$y = \sin(\mu) + \frac{\partial \sin(\mu)}{\partial \mu}(x - \mu) + \ldots$$

which gives

$$\mathbb{E}[y] \approx \mathbb{E}[\sin(\mu) + \cos(\mu)(x - \mu)] = \sin(\mu)$$

$$\text{Cov}[y] \approx \mathbb{E}[(\sin(\mu) + \cos(\mu)(x - \mu) - \sin(\mu))^2] = \cos^2(\mu) \sigma^2.$$
Form 3 sigma points as follows:

\[ \mathcal{X}^{(0)} = \mu \]
\[ \mathcal{X}^{(1)} = \mu + \sigma \]
\[ \mathcal{X}^{(2)} = \mu - \sigma. \]

We may now select some weights \( W^{(0)}, W^{(1)}, W^{(2)} \) such that the original mean and (co)variance can be always recovered by

\[ \mu = \sum_i W^{(i)} \mathcal{X}^{(i)} \]
\[ \sigma^2 = \sum_i W^{(i)} (\mathcal{X}^{(i)} - \mu)^2. \]
Use the same formula for approximating the distribution of $y = \sin(x)$ as follows:

$$
\mu = \sum_i W^{(i)} \sin(x^{(i)})
$$

$$
\sigma^2 = \sum_i W^{(i)} (\sin(x^{(i)}) - \mu)^2.
$$

For vectors $x \sim N(m, P)$ the generalization of standard deviation $\sigma$ is the Cholesky factor $L = \sqrt{P}$:

$$
P = LL^T.
$$

The sigma points can be formed using columns of $L$ (here $c$ is a suitable positive constant):

$$
x^{(0)} = m
$$

$$
x^{(i)} = m + cL_i
$$

$$
x^{(n+i)} = m - cL_i
$$
For transformation \( y = g(x) \) the approximation is:

\[
\mu_y = \sum_i W^{(i)} g(x^{(i)})
\]

\[
\Sigma_y = \sum_i W^{(i)} (g(x^{(i)}) - \mu_y) (g(x^{(i)}) - \mu_y)^T.
\]

Joint distribution of \( x \) and \( y = g(x) + q \) is then given as

\[
E \left[ \begin{pmatrix} x \\ g(x) + q \end{pmatrix} \mid q \right] \approx \sum_i W^{(i)} \begin{pmatrix} x^{(i)} \\ g(x^{(i)}) \end{pmatrix} = \begin{pmatrix} m \\ \mu_y \end{pmatrix}
\]

\[
\text{Cov} \left[ \begin{pmatrix} x \\ g(x) + q \end{pmatrix} \mid q \right] \approx \sum_i W^{(i)} \begin{pmatrix} (x^{(i)} - m)(x^{(i)} - m)^T \\ (g(x^{(i)}) - \mu_y)(x^{(i)} - m)^T \\ (x^{(i)} - m)(g(x^{(i)}) - \mu_y)(x^{(i)} - m)^T \\ (g(x^{(i)}) - \mu_y)(g(x^{(i)}) - \mu_y)(x^{(i)} - m)^T \end{pmatrix}
\]
The unscented transform approximation to the joint distribution of $\mathbf{x}$ and $\mathbf{y} = \mathbf{g}(\mathbf{x}) + \mathbf{q}$ where $\mathbf{x} \sim N(\mathbf{m}, \mathbf{P})$ and $\mathbf{q} \sim N(\mathbf{0}, \mathbf{Q})$ is

$$
\begin{pmatrix}
\mathbf{x} \\
\mathbf{y}
\end{pmatrix} \sim N
\left(
\begin{pmatrix}
\mathbf{m} \\
\mu_U
\end{pmatrix},
\begin{pmatrix}
\mathbf{P} & \mathbf{C}_U \\
\mathbf{C}_U^T & \mathbf{S}_U
\end{pmatrix}
\right),
$$

where the sub-matrices are formed as follows:

Form the sigma points as

$$
\begin{align*}
\chi^{(0)} &= \mathbf{m} \\
\chi^{(i)} &= \mathbf{m} + \sqrt{n+\lambda} \left[ \sqrt{\mathbf{P}} \right]_i \\
\chi^{(i+n)} &= \mathbf{m} - \sqrt{n+\lambda} \left[ \sqrt{\mathbf{P}} \right]_i, \quad i = 1, \ldots, n
\end{align*}
$$
Propagate the sigma points through $g(\cdot)$:

$$\gamma^{(i)} = g(\chi^{(i)}), \quad i = 0, \ldots, 2n.$$ 

The sub-matrices are then given as:

$$\boldsymbol{\mu}_U = \sum_{i=0}^{2n} W_i^{(m)} \gamma^{(i)}$$

$$\boldsymbol{S}_U = \sum_{i=0}^{2n} W_i^{(c)} (\gamma^{(i)} - \boldsymbol{\mu}_U) (\gamma^{(i)} - \boldsymbol{\mu}_U)^T + \mathbf{Q}$$

$$\boldsymbol{C}_U = \sum_{i=0}^{2n} W_i^{(c)} (\chi^{(i)} - \mathbf{m}) (\gamma^{(i)} - \boldsymbol{\mu}_U)^T.$$
Unscented transform (cont.)

- $\lambda$ is a scaling parameter defined as $\lambda = \alpha^2 (n + \kappa) - n$.
- $\alpha$ and $\kappa$ determine the spread of the sigma points.
- Weights $W_i^{(m)}$ and $W_i^{(c)}$ are given as follows:

$$
W_0^{(m)} = \frac{\lambda}{(n + \lambda)}
$$

$$
W_0^{(c)} = \frac{\lambda}{(n + \lambda)} + (1 - \alpha^2 + \beta)
$$

$$
W_i^{(m)} = \frac{1}{2(n + \lambda)}, \quad i = 1, \ldots, 2n
$$

$$
W_i^{(c)} = \frac{1}{2(n + \lambda)}, \quad i = 1, \ldots, 2n,
$$

- $\beta$ can be used for incorporating prior information on the (non-Gaussian) distribution of $\mathbf{x}$. 
\[
\begin{pmatrix}
x_1 \\
x_2
\end{pmatrix}
\sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 2 & -2 \\ -2 & 3 \end{pmatrix}\right)
\]

\[
\frac{dy_1}{dt} = \exp(-y_1), \quad y_1(0) = x_1
\]

\[
\frac{dy_2}{dt} = -\frac{1}{2}y_2^3, \quad y_2(0) = x_2
\]
UT Approximation

Simo Särkkä  Lecture 5: UKF, GF, GHKF and CKF
Assume that the filtering distribution of previous step is Gaussian

\[ p(x_{k-1} \mid y_{1:k-1}) \approx N(x_{k-1} \mid m_{k-1}, P_{k-1}) \]

The joint distribution of \( x_{k-1} \) and \( x_k = f(x_{k-1}) + q_{k-1} \) can be approximated with UT as Gaussian

\[ p(x_{k-1}, x_k, \mid y_{1:k-1}) \approx N \begin{pmatrix} x_{k-1} \\ x_k \end{pmatrix} \mid \begin{pmatrix} m'_1 \\ m'_2 \end{pmatrix}, \begin{pmatrix} P'_{11} & P'_{12} \\ (P'_{12})^T & P'_{22} \end{pmatrix} \]

Form the sigma points \( \chi^{(i)} \) of \( x_{k-1} \sim N(m_{k-1}, P_{k-1}) \) and compute the transformed sigma points as \( \hat{\chi}^{(i)} = f(\chi^{(i)}) \).

The expected values can now be expressed as:

\[ m'_1 = m_{k-1} \]
\[ m'_2 = \sum_i W_i^{(m)} \hat{\chi}^{(i)} \]
The blocks of covariance can be expressed as:

\[ P'_{11} = P_{k-1} \]
\[ P'_{12} = \sum_i W^{(c)}_i (\hat{X}^{(i)} - m_{k-1}) (\hat{X}^{(i)} - m'_2)^T \]
\[ P'_{22} = \sum_i W^{(c)}_i (\hat{X}^{(i)} - m'_2) (\hat{X}^{(i)} - m'_2)^T + Q_{k-1} \]

The prediction mean and covariance of \( x_k \) are then \( m'_2 \) and \( P'_{22} \), and thus we get

\[ m^-_k = \sum_i W^{(m)}_i \hat{X}^{(i)} \]
\[ P^-_k = \sum_i W^{(c)}_i (\hat{X}^{(i)} - m^-_k) (\hat{X}^{(i)} - m^-_k)^T + Q_{k-1} \]
For the joint distribution of $x_k$ and $y_k = h(x_k) + r_k$ we similarly get

$$p(x_k, y_k, | y_{1:k-1}) \approx N \left( \begin{bmatrix} x_k \\ y_k \end{bmatrix}, \begin{bmatrix} m''_1 \\ m''_2 \end{bmatrix}, \begin{bmatrix} P''_{11} & P''_{12} \\ (P''_{12})^T & P''_{22} \end{bmatrix} \right),$$

If $x^{-(i)}$ are the sigma points of $x_k \sim N(m_k^-, P_k^-)$ and $\hat{y}^{(i)} = h(x^{-(i)})$, we get:

- $m''_1 = m_k^-$
- $m''_2 = \sum_i W_i^{(m)} \hat{y}^{(i)}$
- $P''_{11} = P_k^-$
- $P''_{12} = \sum_i W_i^{(c)} (x^{-(i)} - m_k^-) (\hat{y}^{(i)} - m''_2)^T$
- $P''_{22} = \sum_i W_i^{(c)} (\hat{y}^{(i)} - m''_2) (\hat{y}^{(i)} - m''_2)^T + R_k$
Recall that if
\[
\begin{pmatrix} x \\ y \end{pmatrix} \sim N \left( \begin{pmatrix} a \\ b \end{pmatrix}, \begin{pmatrix} A & C \\ C^T & B \end{pmatrix} \right),
\]
then
\[
x | y \sim N(a + CB^{-1}(y - b), A - CB^{-1}C^T).
\]
Thus we get the conditional mean and covariance:
\[
\begin{align*}
m_k &= m_k^- + P_{12}'' (P_{22}'')^{-1}(y_k - m_2'') \\
P_k &= P_k^- - P_{12}''(P_{22}'')^{-1}(P_{12}'')^T.
\end{align*}
\]
Unscented Kalman Filter (UKF): Algorithm [1/4]

Unscented Kalman filter: Prediction step

1. Form the sigma points:

\[
\begin{align*}
\mathcal{X}^{(0)}_{k-1} &= m_{k-1}, \\
\mathcal{X}^{(i)}_{k-1} &= m_{k-1} + \sqrt{n+\lambda} \left[ \sqrt{P_{k-1}} \right]_i, \\
\mathcal{X}^{(i+n)}_{k-1} &= m_{k-1} - \sqrt{n+\lambda} \left[ \sqrt{P_{k-1}} \right]_i, \quad i = 1, \ldots, n.
\end{align*}
\]

2. Propagate the sigma points through the dynamic model:

\[
\hat{\mathcal{X}}^{(i)}_k = f(\mathcal{X}^{(i)}_{k-1}). \quad i = 0, \ldots, 2n.
\]
Compute the predicted mean and covariance:

\[
\begin{align*}
m_k^- &= \sum_{i=0}^{2n} W_i^{(m)} \hat{X}_k^{(i)} \\
P_k^- &= \sum_{i=0}^{2n} W_i^{(c)} (\hat{X}_k^{(i)} - m_k^-) (\hat{X}_k^{(i)} - m_k^-)^T + Q_{k-1}.
\end{align*}
\]
### Unscented Kalman filter: Update step

1. **Form the sigma points:**

   \[
   \chi_k^{-(0)} = m_k^-,
   \]
   \[
   \chi_k^{-(i)} = m_k^- + \sqrt{n + \lambda} \left[ \sqrt{P_k^-} \right]_i, \quad i = 1, \ldots, n.
   \]
   \[
   \chi_k^{-(i+n)} = m_k^- - \sqrt{n + \lambda} \left[ \sqrt{P_k^-} \right]_i, \quad i = 1, \ldots, n.
   \]

2. **Propagate sigma points through the measurement model:**

   \[
   \hat{y}_k^{(i)} = h(\chi_k^{-(i)}), \quad i = 0, \ldots, 2n.
   \]
Unscented Kalman filter: Update step (cont.)

3. Compute the following:

\[ \mu_k = \sum_{i=0}^{2n} W_i^{(m)} \hat{Y}_k^{(i)} \]

\[ S_k = \sum_{i=0}^{2n} W_i^{(c)} (\hat{Y}_k^{(i)} - \mu_k) (\hat{Y}_k^{(i)} - \mu_k)^T + R_k \]

\[ C_k = \sum_{i=0}^{2n} W_i^{(c)} (\hat{X}_k^{(i)} - m_k^-) (\hat{Y}_k^{(i)} - \mu_k)^T \]

\[ K_k = C_k S_k^{-1} \]

\[ m_k = m_k^- + K_k [y_k - \mu_k] \]

\[ P_k = P_k^- - K_k S_k K_k^T. \]
- **No closed form** derivatives or expectations needed.
- **Not a local** approximation, but based on values on a larger area.
- Functions \( f \) and \( h \) do **not** need to be differentiable.
- Theoretically, captures **higher order moments** of distribution than linearization.
Unscented Kalman Filter (UKF): Disadvantage

- **Not a truly global** approximation, based on a small set of trial points.
- Does not work well with nearly **singular covariances**, i.e., with nearly deterministic systems.
- Requires **more computations** than EKF or SLF, e.g., Cholesky factorizations on every step.
- Can only be applied to models driven by **Gaussian noises**.
Consider the transformation of $x$ into $y$:

$$x \sim N(m, P)$$

$$y = g(x).$$

Form Gaussian approximation to $(x, y)$ by directly approximating the integrals:

$$\mu_M = \int g(x) \, N(x \mid m, P) \, dx$$

$$S_M = \int (g(x) - \mu_M)^T (g(x) - \mu_M) \, N(x \mid m, P) \, dx$$

$$C_M = \int (x - m)^T (g(x) - \mu_M)^T \, N(x \mid m, P) \, dx.$$
Gaussian Moment Matching

The moment matching based Gaussian approximation to the joint distribution of $\mathbf{x}$ and the transformed random variable $\mathbf{y} = \mathbf{g}(\mathbf{x}) + \mathbf{q}$ where $\mathbf{x} \sim \mathcal{N}(\mathbf{m}, \mathbf{P})$ and $\mathbf{q} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ is given as

$$
\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \sim \mathcal{N} \left( \begin{bmatrix} \mathbf{m} \\ \mu_{\mathbf{M}} \end{bmatrix}, \begin{bmatrix} \mathbf{P} & \mathbf{C}_{\mathbf{M}} \\ \mathbf{C}_{\mathbf{M}}^T & \mathbf{S}_{\mathbf{M}} \end{bmatrix} \right),
$$

where

$$
\mu_{\mathbf{M}} = \int \mathbf{g}(\mathbf{x}) \mathcal{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P}) \, d\mathbf{x}
$$

$$
\mathbf{S}_{\mathbf{M}} = \int (\mathbf{g}(\mathbf{x}) - \mu_{\mathbf{M}}) (\mathbf{g}(\mathbf{x}) - \mu_{\mathbf{M}})^T \mathcal{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P}) \, d\mathbf{x} + \mathbf{Q}
$$

$$
\mathbf{C}_{\mathbf{M}} = \int (\mathbf{x} - \mathbf{m}) (\mathbf{g}(\mathbf{x}) - \mu_{\mathbf{M}})^T \mathcal{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P}) \, d\mathbf{x}.
$$
Gaussian filter prediction

Compute the following Gaussian integrals:

\[ m_k^- = \int f(x_{k-1}) \, N(x_{k-1} \mid m_{k-1}, P_{k-1}) \, dx_{k-1} \]

\[ P_k^- = \int (f(x_{k-1}) - m_k^-) (f(x_{k-1}) - m_k^-)^T \times N(x_{k-1} \mid m_{k-1}, P_{k-1}) \, dx_{k-1} + Q_{k-1}. \]
Gaussian filter update

1. Compute the following Gaussian integrals:

\[ \mu_k = \int h(x_k) \, N(x_k \mid m_k^-, P_k^-) \, dx_k \]

\[ S_k = \int (h(x_k) - \mu_k) (h(x_k) - \mu_k)^T \, N(x_k \mid m_k^-, P_k^-) \, dx_k + R_k \]

\[ C_k = \int (x_k - m_k^-) (h(x_k) - \mu_k)^T \, N(x_k \mid m_k^-, P_k^-) \, dx_k. \]

2. Then compute the following:

\[ K_k = C_k \, S_k^{-1} \]

\[ m_k = m_k^- + K_k (y_k - \mu_k) \]

\[ P_k = P_k^- - K_k \, S_k \, K_k^T. \]
- Special case of assumed density filtering (ADF).
- Multidimensional Gauss-Hermite quadrature ⇒ Gauss Hermite Kalman filter (GHKF).
- Cubature integration ⇒ Cubature Kalman filter (CKF).
- Monte Carlo integration ⇒ Monte Carlo Kalman filter (MCKF).
- Gaussian process / Bayes-Hermite Kalman filter: Form Gaussian process regression model from set of sample points and integrate the approximation.
- Linearization, unscented transform, central differences, divided differences can be considered as special cases.
One-dimensional Gauss-Hermite quadrature of order $p$:

$$\int_{-\infty}^{\infty} g(x) \ N(x \mid 0, 1) \ dx \approx \sum_{i=1}^{p} W^{(i)} g(x^{(i)}) ,$$

$\xi^{(i)}$ are roots of $p$th order Hermite polynomial:

- $H_0(x) = 1$
- $H_1(x) = x$
- $H_2(x) = x^2 - 1$
- $H_3(x) = x^3 - 3x \ldots$

The weights are $W^{(i)} = p!/(p^2 [H_{p-1}(\xi^{(i)})]^2)$. 

Exact for polynomials up to order $2p - 1$. 

Simo Särkkä Lecture 5: UKF, GF, GHKF and CKF
Multidimensional integrals can be approximated as:

\[
\int g(x) \, N(x | m, P) \, dx = \int g(m + \sqrt{P} \, \xi) \, N(\xi | 0, I) \, d\xi \\
= \int \cdots \int g(m + \sqrt{P} \, \xi) \, N(\xi_1 | 0, 1) \, d\xi_1 \times \cdots \times N(\xi_n | 0, 1) \, d\xi_n \\
\approx \sum_{i_1, \ldots, i_n} W^{(i_1)} \times \cdots \times W^{(i_n)} g(m + \sqrt{P} \, \xi^{(i_1, \ldots, i_n)}).
\]

Needs \(p^n\) evaluation points.

Gauss-Hermite Kalman filter (GHKF) uses this for evaluation of the Gaussian integrals.
Postulate symmetric integration rule:

\[
\int g(\xi) \, N(\xi \mid 0, I) \, d\xi \approx W \sum_i g(c \, u^{(i)}),
\]

where the points \( u^{(i)} \) belong to the symmetric set [1] with generator \((1, 0, \ldots, 0)\):

\[
[1] = \left\{ \left( \begin{array}{c} 1 \\ 0 \\ \vdots \\ 0 \end{array} \right), \left( \begin{array}{c} 0 \\ 1 \\ \vdots \\ 0 \end{array} \right), \cdots, \left( \begin{array}{c} -1 \\ 0 \\ \vdots \\ 0 \end{array} \right), \left( \begin{array}{c} 0 \\ -1 \\ \vdots \\ 0 \end{array} \right), \cdots \right\}
\]

and \( W \) is a weight and \( c \) is a parameter yet to be determined.
Due to symmetry, all odd orders integrated exactly.

We only need to match the following moments:

\[ \int N(\xi \mid 0, I) \, d\xi = 1 \]

\[ \int \xi_j^2 \, N(\xi \mid 0, I) \, d\xi = 1 \]

Thus we get the equations

\[ W \sum_i 1 = W \cdot 2n = 1 \]

\[ W \sum_i [c \, u_j^{(i)}]^2 = W \cdot 2c^2 = 1 \]

Thus the following rule is exact up to third degree:

\[ \int g(\xi) \, N(\xi \mid 0, I) \, d\xi \approx \frac{1}{2n} \sum_i g(\sqrt{n} \, u^{(i)}) \].
General Gaussian integral rule:

\[
\int g(x) \, N(x \mid m, P) \, dx = \int g(m + \sqrt{P} \, \xi) \, N(\xi \mid 0, I) \, d\xi \\
\approx \frac{1}{2n} \sum_{i=1}^{2n} g(m + \sqrt{P} \, \xi^{(i)}),
\]

where

\[
\xi^{(i)} = \begin{cases} 
\sqrt{n} e_i, & i = 1, \ldots, n \\
-\sqrt{n} e_{i-n}, & i = n + 1, \ldots, 2n,
\end{cases}
\]

(1)

where \( e_i \) denotes a unit vector to the direction of coordinate axis \( i \).
Cubature Kalman filter: Prediction step

1. Form the sigma points as:

\[ \chi_{k-1}^{(i)} = m_{k-1} + \sqrt{P_{k-1}} \xi^{(i)} \quad i = 1, \ldots, 2n. \]

2. Propagate the sigma points through the dynamic model:

\[ \hat{\chi}_k^{(i)} = f(\chi_{k-1}^{(i)}), \quad i = 1 \ldots 2n. \]

3. Compute the predicted mean and covariance:

\[
\begin{align*}
\hat{m}_k^- &= \frac{1}{2n} \sum_{i=1}^{2n} \hat{\chi}_k^{(i)} \\
\hat{P}_k^- &= \frac{1}{2n} \sum_{i=1}^{2n} (\hat{\chi}_k^{(i)} - \hat{m}_k^-)(\hat{\chi}_k^{(i)} - \hat{m}_k^-)^T + Q_{k-1}.
\end{align*}
\]

Simo Särkkä
Lecture 5: UKF, GF, GHKF and CKF
Form the sigma points:

\[ \chi_k^{-(i)} = m_k^- + \sqrt{P_k^-} \xi(i), \quad i = 1, \ldots, 2n. \]

Propagate sigma points through the measurement model:

\[ \hat{y}_k^{(i)} = h(\chi_k^{-(i)}), \quad i = 1 \ldots 2n. \]
Cubature Kalman filter: Update step (cont.)

3. Compute the following:

\[
\mu_k = \frac{1}{2n} \sum_{i=1}^{2n} \hat{Y}_k^{(i)}
\]

\[
S_k = \frac{1}{2n} \sum_{i=1}^{2n} (\hat{Y}_k^{(i)} - \mu_k) (\hat{Y}_k^{(i)} - \mu_k)^T + R_k
\]

\[
C_k = \frac{1}{2n} \sum_{i=1}^{2n} (X_k^{(i)} - m_k) (\hat{Y}_k^{(i)} - \mu_k)^T
\]

\[
K_k = C_k S_k^{-1}
\]

\[
m_k = m_k^- + K_k [y_k - \mu_k]
\]

\[
P_k = P_k^- - K_k S_k K_k^T.
\]
Cubature Kalman filter (CKF) is a special case of UKF with $\alpha = 1$, $\beta = 0$, and $\kappa = 0$ – the mean weight becomes zero with these choices.

Rule is exact for third order polynomials (multinomials) – note that third order Gauss-Hermite is exact for fifth order polynomials.

UKF was also originally derived using similar way, but is a bit more general.

Very easy algorithm to implement – quite good choice of parameters for UKF.
Unscented transform (UT) approximates transformations of Gaussian variables by propagating sigma points through the non-linearity.

In UT the mean and covariance are approximated as linear combination of the sigma points.

The unscented Kalman filter uses unscented transform for computing the approximate means and covariance in non-linear filtering problems.

A non-linear transformation can also be approximated with Gaussian moment matching.

Gaussian filter is based on matching the moments with numerical integration ⇒ many kinds of Kalman filters.
Gauss-Hermite Kalman filter (GHKF) uses multi-dimensional Gauss-Hermite for approximation of Gaussian filter.

Cubature Kalman filter (CKF) uses spherical cubature rule for approximation of Gaussian filter – but turns out to be special case of UKF.

We can also use Gaussian processes, Monte Carlo or other methods for approximating the Gaussian integrals.

Taylor series, statistical linearization, central differences and many other methods can be seen as approximations to Gaussian filter.
Recall the discretized pendulum model

\[
\begin{pmatrix}
    x^1_k \\
    x^2_k
\end{pmatrix} =
\begin{pmatrix}
    x^1_{k-1} + x^2_{k-1} \Delta t \\
    x^2_{k-1} - g \sin(x^1_{k-1}) \Delta t
\end{pmatrix} +
\begin{pmatrix}
    0 \\
    q_{k-1}
\end{pmatrix}
\]

\[f(x_{k-1})\]

\[y_k = \sin(x^1_k) + r_k,\]

\[h(x_k)\]

\[\Rightarrow \text{Matlab demonstration}\]