8 Nonlinear Filters

In many applications the basic system we consider is nonlinear, and cannot be accurately described by a linear model. Nonlinearity can arise both in the state dynamics and in the measurement equation. Alternatively, the noise statistics may be complex (e.g., highly non-unimodal), so that a Gaussian or 2nd order model is not adequate to describe it. In all these cases we need to extend the basic filter to accommodate such non-linear effects.

Consider the general state-space model:

\[
\begin{align*}
    x_{k+1} &= f_k(x_k, w_k), \quad k \geq 0 \\
    z_k &= h_k(x_k, v_k),
\end{align*}
\]

where \( \{v_k, w_k\} \) are white noise sequences under the “usual” (strict sense) independence assumptions, with known distributions. The initial state distribution \( p(x_0) \) is given as well.

As before, we wish to estimate \( x_k \) based on \( Z_k = (z_0, \ldots, z_k) \).

Consider first the optimal (MMSE) estimator,

\[
\hat{x}_k = E(x_k|Z_k).
\]

Unfortunately, in general there is no simple way to compute \( \hat{x}_k \), without first computing the entire distribution \( p(x_k) \). Consequently, some approximations are required. Furthermore, in certain cases a mere point estimator \( \hat{x}_k \) cannot adequately describe the state distribution.
The most common current approaches to state estimation in nonlinear models are:

1. The extended Kalman filter
2. Sigma-point Filters (aka Unscented Kalman Filter)
3. Particle Filters

The first two are Kalman-like filters, that are based on linearization of the system around the estimated trajectory. The third approach tries to approximate the entire distribution \( p(x_k) \) using simulation (Monte-Carlo) techniques. We will describe these filters in turn.

8.1 The State Evolution Equations

Let us first write the equations for computation of the conditional distribution \( p(x_k \mid Z_k) \). This computation can be done recursively, using the following two stages:

1. \textit{Time Update}. Compute \( p(x_{k+1} \mid Z_k) \) from \( p(x_k \mid Z_k) \):

\[
p(x_{k+1} \mid Z_k) = \int p(x_{k+1} \mid Z_k, x_k) p(x_k \mid Z_k) \, dx_k
\]

\[
= \int p(x_{k+1} \mid x_k) p(x_k \mid Z_k) \, dx_k,
\]

where \( p(x_{k+1} \mid x_k) \) is induced by the state equation.

2. \textit{Measurement Update}. Compute \( p(x_{k+1} \mid Z_{k+1}) \) from \( p(x_{k+1} \mid Z_k) \):

\[
p(x_{k+1} \mid Z_{k+1}) = p(x_{k+1} \mid Z_k, z_{k+1}) = \frac{p(x_{k+1}, z_{k+1} \mid Z_k)}{p(z_{k+1} \mid Z_k)}
\]

\[
= \frac{p(z_{k+1} \mid x_{k+1}, Z_k)p(x_{k+1} \mid Z_k)}{\int (\cdot - \mu_{Z_k}) \, dx_{k+1}}
\]

\[
= \frac{p(z_{k+1} \mid x_{k+1})p(x_{k+1} \mid Z_k)}{\int (\cdot - \mu_{Z_k}) \, dx_{k+1}},
\]

where \( p(Z_k \mid x_k) \) is induced by the measurement equation.

Given \( p(x_k \mid Z_k) \), we can obviously compute

\[
\hat{x}_k = E(x_k \mid Z_k) = \int x_k p(x_k \mid Z_k) \, dx_k
\]

2
its MSE $E(||x_k - \hat{x}_k||^2|Z_k)$, etc. However, the computation of $p(x_k|Z_k)$ using numerical integration techniques is complex, and becomes unfeasible if $x$ is high-dimensional.

We therefore seek approximate solutions.

8.2 Observer-Like Filters

Motivated by the linear case, we will first look for sub-optimal filters of the form:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(z_k - \overline{h}_k(\hat{x}_{k|k-1}))$$
$$\hat{x}_{k+1|k} = \overline{f}_k(\hat{x}_{k|k})$$

where

$$\overline{f}_k(x_k) = E(f_k(x_k, w_k) | x_k)$$
$$\overline{h}_k(x_k) = E(h_k(x_k, v_k) | x_k).$$

Note that in the common model:

$$x_{k+1} = f(x_k) + g(x_k)w_k$$
$$z_k = h(x_k) + i(x_k)v_k$$

(where $w_k$ and $v_k$ are zero-mean), we simply have

$$\overline{f}(x_k) = f(x_k),$$
$$\overline{h}(x_k) = h(x_k).$$

The problem still remains: How to choose $K_k$, and how to estimate the covariance $P_k$.

For that purpose we can use linearization.
8.3 Linearization

The linearized model at time $k$ around some state $x^0_k$ (and 0 noise levels) is

$$
x_{k+1} \simeq f(x^0_k, 0) + F_k(x_k - x^0_k) + G_k w_k
$$

$$
z_k \simeq h(x^0_k, 0) + H_k(x_k - x^0_k) + I_k v_k,
$$

where

$$
F_k = \left. \frac{\partial f_k(x, w)}{\partial x} \right|_{(x^0_k, 0)}
$$

$$
G_k = \left. \frac{\partial f_k(x, w)}{\partial w} \right|_{(x^0_k, 0)}
$$

$$
H_k = \left. \frac{\partial h_k(x, v)}{\partial x} \right|_{(x^0_k, 0)}
$$

$$
I_k = \left. \frac{\partial h_k(x, v)}{\partial v} \right|_{(x^0_k, 0)}
$$

and all derivatives are evaluated at the point $x = x^0_k, w = 0, v = 0$. This gives a linear system (+ bias), and we can compute the KF gain and covariance with respect to $(F_k, G_k, H_k, I_k)$.

We still need to determine $x^0_k$. The ideal choice would be $x^0_k = x_k$ (the actual system state). However, $x_k$ is unknown.

The following choices are feasible:

(i) **Static linearization:** Choose some fixed nominal value: $x^0_k \equiv x^0$. This may work well if $\{F, G, H, I\}$ depend weakly on $x$.

(ii) **Predefined trajectory:** If an approximate nominal trajectory ($x^*_k$) is known beforehand, we can use $x^0_k = x^*_k$.

(iii) **The current estimate:** Choose $x^0_k$ as the most recent estimate of $x_k$, namely $x^0_k \triangleq \hat{x}_{k|k-1}$ or $\hat{x}_{k|k}$. This gives rise to the extended KF.
8.4 The Extended Kalman Filter (EKF)

The EKF proceeds as follows:

▷ Start with $\hat{x}_{k|k-1}$ and $P_{k|k-1}$.

▷ Measurement Update ($\hat{x}_{k|k-1} \rightarrow \hat{x}_{k|k}$):
  ◦ Compute $H_k$ and $I_k$ at $\hat{x}_{k|k-1}$.
  ◦ $\tilde{z}_k = z_k - h_k(\hat{x}_{k|k-1})$.
  ◦ Compute $K_k$, $\hat{x}_{k|k}$, $P_{k|k}$, using the usual equations.

▷ Time Update ($\hat{x}_{k|k} \rightarrow \hat{x}_{k+1|k}$):
  ◦ Compute $F_k$ and $G_k$ at $\hat{x}_{k|k}$.
  ◦ $\hat{x}_{k+1|k} = f_k(\hat{x}_{k|k})$.
  ◦ $P_{k+1|k} = F_k P_{k|k} F_k^T + G_k Q_k G_k^T$.

Remarks:

1. The EKF is a heuristic filter based on first-order approximation. It is not optimal, and no guarantee of bounded error is provided a-priori.

2. The EKF is non-stationary. Its gain $K_k$ and covariance $P_k$ must be computed on-line.

3. The computed covariance matrix $P_k$ is no longer the actual error covariance, but rather an estimate.
   It might happen, for example, that $P_k$ is small while the actual error is large; this causes “filter divergence”.

4. The EKF may fail completely in highly nonlinear problems. However, in many applications it performs well, after some tuning.

5. The covariances $Q_k$, $R_k$ should be taken large enough to “cover” also for linearization errors.
The Iterated EKF: A simple improvement of the EKF is obtained by repeating the linearization step with the new value of $\hat{x}_k$. This applies to the measurement update, as follows:

▷ start with $\hat{x}^0_{k|k} = \hat{x}_{k|k-1}$.

▷ for $i = 1, \ldots, N$:
  - compute $H^i_k$, $I^i_k$ and $\tilde{z}^i_k$ at $\hat{x}^i_{k|k}$
  - compute $K^{i+1}_k$, $\hat{x}^{i+1}_{k|k}$, $P^{i+1}_k$.

$N$ may be fixed as a small integer (say 4), or the iteration may continue until the difference is small.
The Unscented Kalman Filter (UKF)

The Unscented KF takes another approach to linearization – which uses representative sample points. It was shown in various applications to be more robust than the EKF, while requiring the same order of calculations. Moreover, it does not require analytic computation of the derivatives of the nonlinearities, and not even that these derivatives exist.

Original References:

a. Estimating nonlinear functions of random vectors

Let \( \mathbf{x} \) be a random vector with known mean \( \mathbf{m}_x \) and covariance \( \mathbf{P}_x \). Let \( \mathbf{y} = f(\mathbf{x}) \) be a nonlinear transformation of \( \mathbf{x} \), and suppose we want to estimate the mean \( \mathbf{m}_y \) and covariance \( \mathbf{P}_y \) of \( \mathbf{y} \). Possible approaches include the following:

1. Local linearization: Let \( F = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \bigg|_{\mathbf{x} = \mathbf{m}_x} \). Then \( \mathbf{y} \approx F(\mathbf{x} - \mathbf{m}_x) + f(\mathbf{m}_x) \), hence \( \hat{\mathbf{m}}_y = f(\mathbf{m}_x) \) and \( \hat{\mathbf{P}}_y = F\mathbf{P}_x F^T \).

   This approach leads to the Extended KF.

2. Monte-Carlo sampling: Here we assume that the distribution \( p_x \) of \( \mathbf{x} \) is given. We sample random points \( \{ \mathcal{X}_1, \ldots, \mathcal{X}_N \} \) from \( p_x \), and compute \( \mathcal{Y}_i = f(\mathcal{X}_i) \) for \( i = 1, \ldots, N \). We can new estimate:

   \[
   \hat{\mathbf{m}}_y = \frac{1}{N} \sum_{i=1}^{N} \mathcal{Y}_i, \quad \hat{\mathbf{P}}_y = \frac{1}{N} \sum_{i=1}^{N} (\mathcal{Y}_i - \hat{\mathbf{m}}_y)(\mathcal{Y}_i - \hat{\mathbf{m}}_y)^T .
   \]

   In fact, we can use \( \{ \mathcal{Y}_1, \ldots, \mathcal{Y}_N \} \) to estimate the entire distribution \( p_y \).

   This approach leads the the Particle Filter that we describe later.
3. Deterministic sample points: Here we choose deterministic points \( \{X_1, \ldots, X_N\} \) and corresponding weights \( \{W_1, \ldots, W_N\} \) with \( \sum_{i=1}^{N} W_i = 1 \) so that:

\[
  m_x = \sum_{i=1}^{N} W_i X_i, \quad P_x = \sum_{i=1}^{N} W_i (X_i - m_x)(X_i - m_x)^T.
\]

We can now compute \( Y_i = f(X_i) \) and estimate

\[
  \hat{m}_y = \sum_{i=1}^{N} W_i Y_i, \quad \hat{P}_y = \sum_{i=1}^{N} W_i (Y_i - \hat{m}_y)(Y_i - \hat{m}_y)^T.
\]

The points \( \{X_1, \ldots, X_N\} \) are called sigma points.

This approach leads to the UKF.

b. The Unscented Transform

We now specify the choice of the sigma points. Assuming \( x \in \mathbb{R}^n \), let

\[
  X_0 = m_x, \quad W_0 = \frac{\lambda}{n + \lambda} \\
  X_i = m_x + \sqrt{n + \lambda}(\sqrt{P_x})_i, \quad W_i = \frac{1}{2(n + \lambda)}, \quad i = 1 \ldots n \\
  X_{n+i} = m_x - \sqrt{n + \lambda}(\sqrt{P_x})_i, \quad W_{n+i} = \frac{1}{2(n + \lambda)}, \quad i = 1 \ldots n
\]

Here \( \sqrt{P} \) is the square root of \( P \), namely any matrix \( M \) so that \( MM^T = P \), and \( (\sqrt{P})_i \) is its \( i \)-th column. \( \lambda \) is a (positive of negative) parameter that controls the weight of \( X_0 \).

It is easily seen that this choice satisfies the required equalities for \( m_x \) and \( P_x \). Furthermore, the odd central moments are all zero by symmetry.

Remarks:

The parameter \( \lambda \) controls the separation of the sigma points, can be tuned (for example) to match 4th moments of \( p_x \). For the Gaussian distribution, this is obtained for \( n + \lambda = 3 \). (Verify that for \( P_x = I \) this choice obtains \( E(X_i^4) = 3. \) However, note that \( \lambda < 0 \) may lead to loss of positive definiteness of covariance matrices (see below), hence should be handled with care.
Efficient and robust algorithms are available to compute $\sqrt{P}$. In particular, the Cholesky decomposition gives a lower-triangular $\sqrt{P}$.

The estimates $\hat{m}_y$ and $\hat{P}_y$ are computed as in equation (1).

c. Basic filter equations

Let us return to the dynamic system $x_{k+1} = f_k(x_k, w_k)$, $z_k = h_k(x_k, v_k)$. At stage $k$, we start with $\hat{x}_{k-1|k-1}$ and $P_{k-1|k-1}$, which we consider as estimates for the first and second moments of the random vector $x_{k-1|k-1}$. The filter equations that we wish to approximate are:

(i) Time update:

$$\hat{x}_{k|k-1} = E(x_{k|k-1}), \quad P_{k|k-1} = \text{cov}(x_{k|k-1})$$

where

$$x_{k|k-1} = f_{k-1}(x_{k-1|k-1}, w_{k-1})$$

(ii) Measurement update:

$$\hat{x}_{k} = \hat{x}_{k|k-1} + K_k \tilde{z}_k$$

$$P_{k|k} = P_{k|k-1} - K_k \text{cov}(\tilde{z}_k)K_k^T$$

where

$$\tilde{z}_k = z_k - \hat{z}_{k|k-1}$$

$$\hat{z}_{k|k-1} = E(h_k(x_{k|k-1}, v_k))$$

$$K_k = \text{cov}(x_k, z_k)\text{cov}(\tilde{z}_k)^{-1}$$

d. Sigma points for stage $k$

Let

$$x^a = \begin{pmatrix} x_{k-1|k-1} \\ w_{k-1} \\ v_k \end{pmatrix}, \quad P^a = \text{cov}(x_a) = \begin{pmatrix} P_{k-1|k-1} & 0 & 0 \\ 0 & Q_{k-1} & 0 \\ 0 & 0 & R_k \end{pmatrix}$$

Choose sigma points $X^a_0, \ldots, X^a_N$ for $x^a$ (where $N = 2(n_x + n_w + n_v)$), with weights $W_0, \ldots, W_N$. 

9
Denote (using Matlab notation) $\mathcal{X}_i^a = [\mathcal{X}_i^{k-1|k-1}, \mathcal{X}_i^w, \mathcal{X}_i^v]$.

**e. Time Update**

Compute

$$\mathcal{X}_i^{k|k-1} = f_{k-1}(\mathcal{X}_i^{k-1|k-1}, \mathcal{X}_i^w), \quad i = 0, \ldots, N$$

and

$$\hat{x}_{k|k-1} = \sum_{i=0}^{N} W_i \mathcal{X}_i^{k|k-1}$$

$$P_{k|k-1} = \sum_{i=0}^{N} W_i (\mathcal{X}_i^{k|k-1} - \hat{x}_{k|k-1})(\mathcal{X}_i^{k|k-1} - \hat{x}_{k|k-1})^T.$$  

**f. Measurement Update**

Compute

$$\mathcal{Z}_i^{k|k-1} = h_{k-1}(\mathcal{X}_i^{k|k-1}, \mathcal{X}_i^v), \quad i = 0, \ldots, N$$

$$\hat{z}_{k|k-1} = \sum_{i=0}^{N} W_i \mathcal{Z}_i^{k|k-1}$$

$$\text{cov}(\hat{z}_k) = \sum_{i=0}^{N} W_i (\mathcal{Z}_i^{k|k-1} - \hat{z}_{k|k-1})(\mathcal{Z}_i^{k|k-1} - \hat{z}_{k|k-1})^T$$

$$\text{cov}(x_k, z_k) \equiv \text{cov}(x_{k|k-1}, z_{k|k-1})$$

$$= \sum_{i=0}^{N} W_i (\mathcal{X}_i^{k|k-1} - \hat{x}_{k|k-1})(\mathcal{Z}_i^{k|k-1} - \hat{z}_{k|k-1})^T$$

and substitute in the filter equations:

$$K_k = \text{cov}(x_k, z_k)\text{cov}(\hat{z}_k)^{-1}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(z_k - \hat{z}_{k|k-1})$$

$$P_{k|k} = P_{k|k-1} - K_k\text{cov}(\hat{z}_k)K_k^T.$$