Kalman Filter

LIN Qingfeng
March 3, 2021

1 Introduction

- Filtering is desirable in many situation in engineering. A good filtering algorithm can remove the noise while retaining the useful information. **The Kalman filter** is a tool that can estimate the variables of a wide range of processes. In mathematical terms we can say that a Kalman filter estimates the state of a linear system. The Kalman filter not only works well in practice, but it is theoretically attractive because it can be shown that of all possible filters, it is the one that **minimizes the variance of the estimation error**.
- Accurate system models are not as readily available for industrial problems. In addition, it is difficult to acquire the statistical nature of the noise processes that impinge on such systems. We needed a new filter that could handle system modeling errors and noise uncertainty. State estimators that can tolerate such uncertainty are called robust. The H_∞ filter was designed for robustness.
- The standard Kalman filter is an effective tool for estimation, but it is limited to linear systems. Most real-world systems are nonlinear, in which case Kalman filters do not directly apply. In the real world, nonlinear filters are used more often than linear filters, because in the real world, systems are nonlinear. Thus, we need a new filter to estimate the states in nonlinear system.

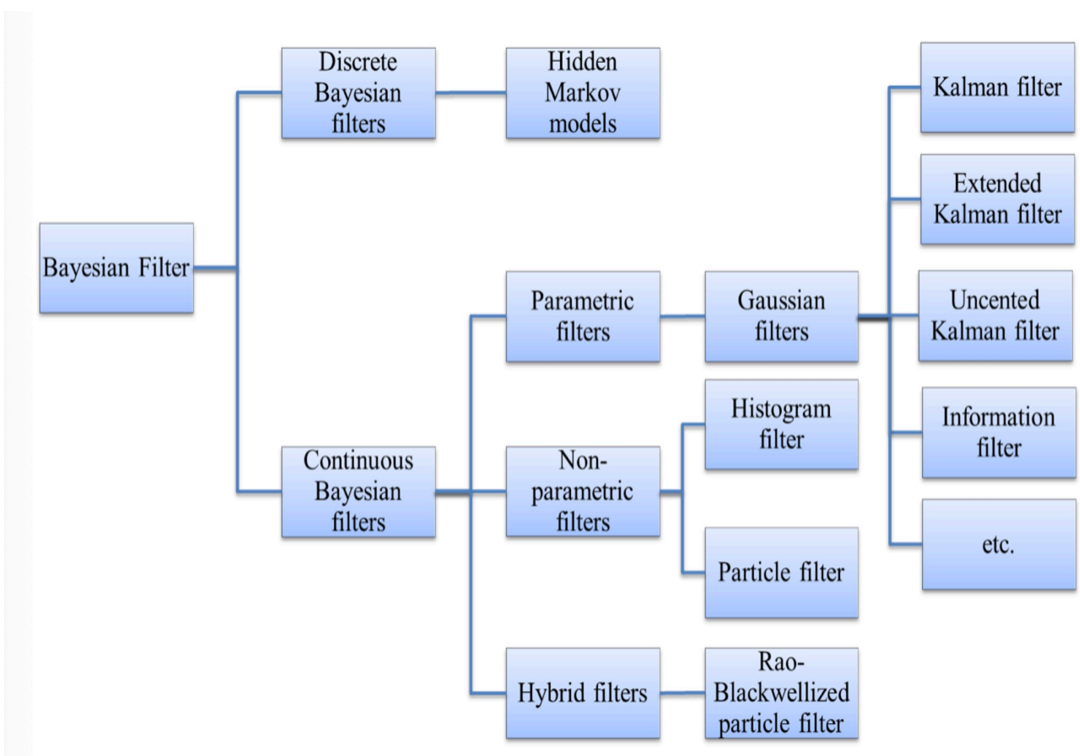

Figure 1: Bayesian Filter

2 Some Basics of Concept

- **States**: The states of a system are those variables that provide a complete representation of the internal condition or status of the system at a given instant of time.
- **A posteriori estimate**: estimate of x_k after we process the measurement at time k

$$\hat{x}_k^+ = E\left[x_k \mid y_1, y_2, \cdots, y_k\right] \tag{1}$$

- **A priori estimate**: estimate of x_k before we process the measurement at time k

$$\hat{x}_k^- = E\left[x_k \mid y_1, y_2, \cdots, y_{k-1}\right] \tag{2}$$

- **Initial state**: using $\hat{x}_0^+$ to denote our initial estimate of x_0 before any measurements are available.

$$\hat{x}_0^+ = E\left(x_0\right) \tag{3}$$

- **The covariance of the estimation error of $\hat{x}_k^-$:**

$$P_k^- = E\left[\left(x_k - \hat{x}_k^-\right)\left(x_k - \hat{x}_k^-\right)^T\right] \tag{4}$$

- **The covariance of the estimation error of $\hat{x}_k^+$:**

$$P_k^+ = E\left[\left(x_k - \hat{x}_k^+\right)\left(x_k - \hat{x}_k^+\right)^T\right] \tag{5}$$

- **Robust:** State estimators that can tolerate such uncertainty are called robust.

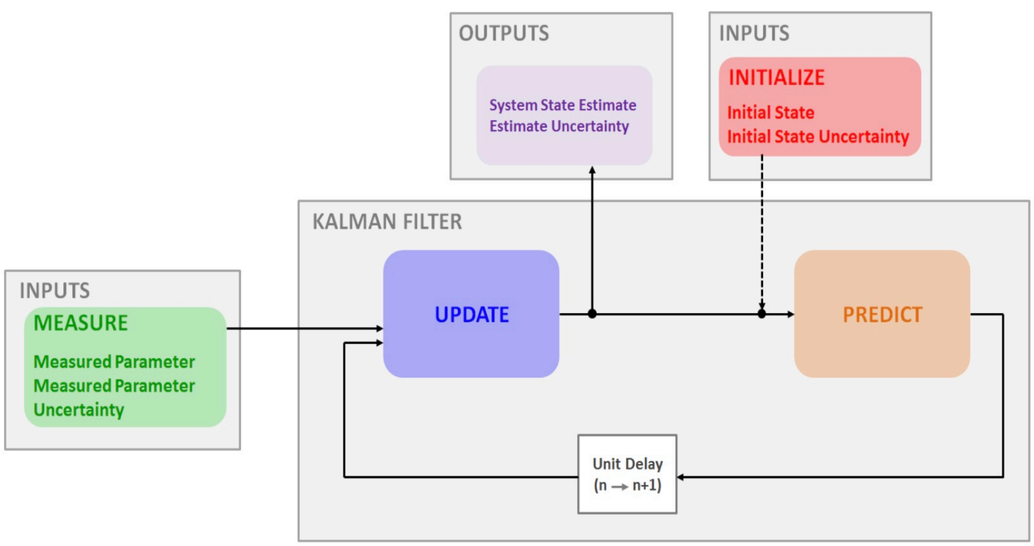

Figure 2: The Algorithm of Kalman Filter

3 Least Square Estimation

Least Square is an important estimation method and it will enable us to derive optimal state estimators.

3.1 Estimation of a constant

- The model is,

$$y = Hx + v \tag{6}$$

where y is measurement data and x is the variable we need estimate.

Now define ϵ_y as the difference between the noisy measurements and the vector $H\hat{x}$

$$\epsilon_y = y - H\hat{x} \tag{7}$$

ϵ_y is called the measurement residual. The cost function is,

$$\begin{aligned} J &= (y - H\hat{x})^T (y - H\hat{x}) \\ &= y^T y - \hat{x}^T H^T y - y^T H\hat{x} + \hat{x}^T H^T H\hat{x} \end{aligned} \tag{8}$$

In order to minimize J with respect to $\hat{x}$, we compute its partial derivative and set it equal to zero:

$$\hat{x} = \left(H^T H\right)^{-1} H^T y \tag{9}$$

3.2 Weighted Least Squares Estimation

- In this equation,

$$J = (y - H\hat{x})^T (y - H\hat{x}) \tag{10}$$

We assumed that we had an equal amount of confidence in all of our measurements. However, in other cases, we need suppose we have more confidence in some measurements than others. Thus, we should generalize the results of the previous section.

We update the cost function,

$$\begin{aligned} J &= \epsilon_y^T R^{-1} \epsilon_y \\ &= (y - H\hat{x})^T R^{-1} (y - H\hat{x}) \\ &= y^T R^{-1} y - \hat{x}^T H^T R^{-1} y - y^T R^{-1} H\hat{x} + \hat{x}^T H^T R^{-1} H\hat{x} \end{aligned} \tag{11}$$

$$\begin{aligned} R &= E\left(vv^T\right) \\ &= \begin{bmatrix} \sigma_1^2 & \cdots & 0 \\ \vdots & & \vdots \\ 0 & \cdots & \sigma_k^2 \end{bmatrix} \end{aligned} \tag{12}$$

Minimize J with respect to $\hat{x}$, we compute its partial derivative and set it equal to zero:

$$\hat{x} = \left(H^T R^{-1} H\right)^{-1} H^T R^{-1} y \tag{13}$$

We should note that this method requires that the measurement noise matrix R be non-singular.

3.3 Recursive Least Squares Estimation

- The equation (7) gives us a way to compute the optimal estimate of a constant. However, there is a problem. If we obtain measurements sequentially and want to update our estimate of x with each new measurement, we need to augment the H matrix and completely recompute the estimate $\hat{x}$. If the number of measurements becomes large, then the computational effort could become prohibitive.
- In order to solve the problem above, we can use a recursive method. A linear recursive estimator can be written in the form,

$$\begin{aligned} y_k &= H_k x + v_k \\ \hat{x}_k &= \hat{x}_{k-1} + K_k \left(y_k - H_k \hat{x}_{k-1}\right) \end{aligned} \tag{14}$$

  That is, we compute $\hat{x}_k$ on the basis of the previous estimate $\hat{x}_{k-1}$ and the new measurement $y_k . K_k$ is a matrix to be determined called the estimator gain matrix. The quantity $\left(y_k - H_k \hat{x}_{k-1}\right)$ is called the correction term.
- Note that if the measurement noise v_k is zero-mean for all k, and the initial estimate of x is set equal to the expected value of x [i.e., $\hat{x}_0 = E(x)$]. Then the estimator of Equation (12) is an unbiased estimator.
- In order to calculate K_k, we should choose a proper cost function which is a optimality criterion. The optimality criterion that we choose to minimize is the sum of the variances of the estimation errors at time k

$$\begin{aligned} J_k &= E\left[\left(\left(x_1 - \hat{x}_1\right)^2\right)\right] + \cdots + E\left[\left(\left(x_n - \hat{x}_n\right)^2\right)\right] \\ &= E\left(\epsilon_{x1,k}^2 + \cdots + \epsilon_{xn,k}^2\right) \\ &= E\left(\epsilon_{x,k}^T \epsilon_{x,k}\right) \\ &= E\left[\operatorname{Tr}\left(\epsilon_{x,k} \epsilon_{x,k}^T\right)\right] \end{aligned} \tag{15}$$

- Minimize J_k with respect to K_k, we compute its derivative and set it equal to zero:

$$K_k = P_{k-1} H_k^T \left(H_k P_{k-1} H_k^T + R_k\right)^{-1} \tag{16}$$

4 Kalman Filter

4.1 The discrete-time Kalman Filter

- **Model:** The dynamic system is given by the following equations:

$$\begin{aligned} x_k &= F_{k-1} x_{k-1} + G_{k-1} u_{k-1} + w_{k-1} \\ y_k &= H_k x_k + v_k \\ E\left(w_k w_j^T\right) &= Q_k \delta_{k-j} \\ E\left(v_k v_j^T\right) &= R_k \delta_{k-j} \\ E\left(w_k v_j^T\right) &= 0 \end{aligned} \tag{17}$$

  The Kalman filter is initialized as follows:

$$\begin{aligned} \hat{x}_0^+ &= E\left(x_0\right) \\ P_0^+ &= E\left[\left(x_0 - \hat{x}_0^+\right)\left(x_0 - \hat{x}_0^+\right)^T\right] \end{aligned} \tag{18}$$

The Kalman filter is given by the following equations, which are computed for each time step $k = 1, 2, \cdots$

$$\begin{aligned}
P_k^- &= F_{k-1} P_{k-1}^+ F_{k-1}^T + Q_{k-1} \\
K_k &= P_k^- H_k^T \left(H_k P_k^- H_k^T + R_k \right)^{-1} \\
&= P_k^+ H_k^T R_k^{-1} \\
\hat{x}_k^- &= F_{k-1} \hat{x}_{k-1}^+ + G_{k-1} u_{k-1} = \text{ a priori state estimate} \\
\hat{x}_k^+ &= \hat{x}_k^- + K_k \left(y_k - H_k \hat{x}_k^- \right) = \text{a posteriori state estimate} \\
P_k^+ &= \left(I - K_k H_k \right) P_k^- \left(I - K_k H_k \right)^T + K_k R_k K_k^T \\
&= \left[\left(P_k^- \right)^{-1} + H_k^T R_k^{-1} H_k \right]^{-1} \\
&= \left(I - K_k H_k \right) P_k^-
\end{aligned} \tag{19}$$

4.2 Sequential Kalman Filter

- **Comments**: Sequential Kalman filter, which is mathematically identical to the Kalman filter, but which **avoids matrix inversion**. However, sequential filtering can only be used if the noise covariance is diagonal, or if the noise covariance is constant.
- **Model**:

$$\begin{aligned}
x_k &= F_{k-1} x_{k-1} + G_{k-1} u_{k-1} + w_{k-1} \\
y_k &= H_k x_k + v_k \\
w_k &\sim (0, Q_k) \\
v_k &\sim (0, R_k)
\end{aligned} \tag{20}$$

where w_k and v_k are uncorrelated white noise sequences. The measurement covariance R_k is a diagonal matrix given as

$$R_k = \text{diag}\left(R_{1k}, \cdots, R_{rk} \right)$$

At each time step k, the measurement-update equations are given as follows.
(a) Initialize the a posteriori estimate and covariance as

$$\begin{aligned}
\hat{x}_{0k}^+ &= \hat{x}_k^- \\
P_{0k}^+ &= P_k^-
\end{aligned} \tag{21}$$

(b) For $i = 1, \cdots, r$ (where r is the number of measurements), perform the following:

$$\begin{aligned}
K_{ik} &= \frac{P_{i-1,k}^+ H_{ik}^T}{H_{ik} P_{i-1,k}^+ H_{ik}^T + R_{ik}} \\
&= \frac{P_{ik}^+ H_{ik}^T}{R_{ik}} \\
\hat{x}_{ik}^+ &= \hat{x}_{i-1,k}^+ + K_{ik} \left(y_{ik} - H_{ik} \hat{x}_{i-1,k}^+ \right) \\
P_{ik}^+ &= \left(I - K_{ik} H_{ik} \right) P_{i-1,k}^+ \left(I - K_{ik} H_{ik} \right)^T + K_{ik} R_{ik} K_{ik}^T
\end{aligned} \tag{22}$$

(c) Assign the a posteriori estimate and covariance as

$$\begin{aligned}
\hat{x}_k^+ &= \hat{x}_{rk}^+ \\
P_k^+ &= P_{rk}^+
\end{aligned} \tag{23}$$

4.3 Information Filtering

- **Comments**: Information filtering is equivalent to the Kalman filter, but it propagates the inverse of the covariance. This can be computationally beneficial in cases in which the number of **measurements is much larger than the number of states**.
- **Model**: The Kalman filter is initialized as follows:

$$
\begin{aligned}
\hat{x}_0^+ &= E\left(x_0\right) \\
\mathcal{I}_0^+ &= \left\{E\left[\left(x_0-\hat{x}_0^+\right)\left(x_0-\hat{x}_0^+\right)^T\right]\right\}^{-1}
\end{aligned}
\tag{24}
$$

The information filter is given by the following equations, which are computed for each time step $k=1,2,\cdots$

$$
\begin{aligned}
&\mathcal{I}_k^- = Q_{k-1}^{-1}-Q_{k-1}^{-1}F_{k-1}\left(\mathcal{I}_{k-1}^+ + F_{k-1}^T Q_{k-1}^{-1}F_{k-1}\right)^{-1}F_{k-1}^T Q_{k-1}^{-1} \\
&\mathcal{I}_k^+ = \mathcal{I}_k^- + H_k^T R_k^{-1} H_k \\
&K_k = \left(\mathcal{I}_k^+\right)^{-1} H_k^T R_k^{-1} \\
&\hat{x}_k^- = F_{k-1}\hat{x}_{k-1}^+ + G_{k-1}u_{k-1} \\
&\hat{x}_k^+ = \hat{x}_k^- + K_k\left(y_k - H_k\hat{x}_k^-\right)
\end{aligned}
\tag{25}
$$

The standard Kalman filter equations require the inversion of an $r \times r$ matrix, where r is the number of measurements. The information filter equations require at least a couple of $n \times n$ matrix inversions, where n is the number of states. Therefore, if $r \gg n$ (i.e., we have significantly more measurements than states) it may be computationally more efficient to use the information filter.

4.4 Square Root Filtering

- **Comments**: Square root filtering effectively increase the precision of the Kalman filter. Although this approach requires additional computational effort, it can help prevent divergence and instability.
- **Model**:

1. After the a priori covariance square root S_k^- and the a priori state estimate $\hat{x}_2^-$ have been computed. initialize

$$
\begin{aligned}
\hat{x}_{0k}^+ &= \hat{x}_k^- \\
S_{0k}^+ &= S_k^-
\end{aligned}
\tag{26}
$$

2. For $i=1,\cdots,r$ (where r is the number of measurements), perform the following.

(a) Define H_{ik} as the i th row of H_k, y_{ik} as the i th element of y_k, and R_{ik} as the variance of the i th measurement (assuming that R_k is diagonal).

(b) Perform the following to find the square root of the covariance after the i th measurement has been processed:

$$
\begin{aligned}
\phi_i &= S_{i-1,k}^{+T} H_{ik}^T \\
a_\imath &= \frac{1}{\phi_i^T\phi_i + R_{ik}} \\
\gamma_i &= \frac{1}{1 \pm \sqrt{a_\imath R_{ik}}} \\
S_{ik}^+ &= S_{i-1,k}^+\left(I - a_i\gamma_i\phi_i\phi_i^T\right)
\end{aligned}
\tag{27}
$$

(c) Compute the Kalman gain for the ith measurement as

$$
K_{ik} = a_2 S_{ik}^+ \phi_i
\tag{28}
$$

(d) Compute the state estimate update due to the i th measurement as

$$\hat{x}_{ik}^{+} = \hat{x}_{i-1,k}^{+} + K_{ik}\left(y_{ik} - H_{ik}\hat{x}_{i-1,k}^{+}\right) \tag{29}$$

3. Set the a posteriori covariance square root and the a posteriori state estimate as

$$\begin{aligned} S_k^{+} &= S_{rk}^{+} \\ \hat{x}_k^{+} &= \hat{x}_{rk}^{+} \end{aligned} \tag{30}$$

5 H-infinity Filter

- **The limitations of Kalman Filter**:
  1. First, we need to know the mean and correlation of the noise w_k and v_k at each time instant. Also, we should know the system model matrices F_k and H_k.
  2. We need to know the covariances Q_k and R_k of the noise processes. The Kalman filter uses Q_k and R_k as design parameters, so if we do not know Q_k and R_k then it may be difficult to successfully use a Kalman filter.
- **The definition of H-infinity Filter:**
  The H_∞, also called the minimax filter.The H_∞ filter does not make any assumptions about the noise, and it minimizes the worst-case estimation error (hence the term minimax).
- The system model are given as,

$$\begin{aligned} x_{k+1} &= F_k x_k + w_k \\ y_k &= H_k x_k + v_k \end{aligned} \tag{31}$$

  where w_k and v_k are noise terms. These noise term may be random with possibly unknown statistics and our goal is to estimate a linear combination of the state. That is, we want to estimate z_k, which is given by

$$z_k = L_k x_k \tag{32}$$

- The cost function is given based on game theory,

$$J_1 = \frac{\sum_{k=0}^{N-1} ||z_k - \hat{z}_k||_{S_k}^2}{||x_0 - \hat{x}_0||_{P_0^{-1}}^2 + \sum_{k=0}^{N-1}\left(||w_k||_{Q_k^{-1}}^2 + ||v_k||_{R_k^{-1}}^2\right)} \tag{33}$$

- The idea is, our goal as engineers is to find an estimate $\hat{z}_k$ that minimizes J_1. Nature's goal as our adversary is to find disturbances w_k and v_k, and the initial state x_0, to maximize J_1.
- The difference is, in Kalman filtering, nature is assumed to be indifferent. The pdf of the noise is given. We (as filter designers) know the pdf of the noise and can use that knowledge to obtain a statistically optimal state estimate. But nature cannot change the pdf to degrade our state estimate. In H_∞ filtering, nature is assumed to be perverse and actively seeks to degrade our state estimate as much as possible.
- The direct minimization of J_1 is not tractable, so instead we choose a performance bound and seek an estimation strategy that satisfies the threshold. That is, we will try to find an estimate $\hat{z}_k$ that results in

$$J_1 < \frac{1}{\theta} \tag{34}$$

- The minimax problem becomes

$$J^* = \min_{\hat{x}_k} \max_{w_k, y_k, x_0} J \tag{35}$$

  That is an optimization problem, we can use proper algorithm to solve this.

6 Nonlinear Kalman Filter

- **Motivation:**
  All of our discussion to this point has considered linear filters for linear systems.Unfortunately, linear systems do not exist. All systems are ultimately nonlinear.Thus, it is necessary to design nonlinear filter.

  Consider the following general nonlinear system model:

$$\begin{aligned} \dot{x} &= f(x, u, w, t) \\ y &= h(x, v, t) \\ w &\sim (0, Q) \\ v &\sim (0, R) \end{aligned} \tag{36}$$

  The system equation $f(\cdot)$ and the measurement equation $h(\cdot)$ are nonlinear functions. We will use **Taylor series** to expand these equations around a nominal control u_0, nominal state x_0, nominal output y_0, and nominal noise values w_0 and v_0.

  The Taylor series linearization gives,

$$\begin{aligned} \dot{x} \approx& f\left(x_0, u_0, w_0, t\right) + \left.\frac{\partial f}{\partial x}\right|_0 \left(x - x_0\right) + \left.\frac{\partial f}{\partial u}\right|_0 \left(u - u_0\right) + \\ & \left.\frac{\partial f}{\partial w}\right|_0 \left(w - w_0\right) \\ =& f\left(x_0, u_0, w_0, t\right) + A\Delta x + B\Delta u + L\Delta w \\ y \approx& \, h\left(x_0, v_0, t\right) + \left.\frac{\partial h}{\partial x}\right|_0 \left(x - x_0\right) + \left.\frac{\partial h}{\partial v}\right|_0 \left(v - v_0\right) \\ =& \, h\left(x_0, v_0, t\right) + C\Delta x + M\Delta v \end{aligned} \tag{37}$$

6.1 Extended Kalman Filter

- **Idea:**
  We linearize the nonlinear system around the Kalman filter estimate, and the Kalman filter estimate is based on the linearized system. This is the idea of the extended Kalman filter (EKF).
- The system equations are given as

$$\begin{aligned} \dot{x} &= f(x, u, w, t) \\ y &= h(x, v, t) \\ w &\sim (0, Q) \\ v &\sim (0, R) \end{aligned} \tag{38}$$

  The nominal trajectory is known ahead of time:

$$\begin{aligned} \dot{x_0} &= f\left(x_0, u_0, 0, t\right) \\ y_0 &= h\left(x_0, 0, t\right) \end{aligned} \tag{39}$$

- Compute the following partial derivative matrices evaluated at the nominal trajectory values:

$$
\begin{aligned}
A &= \left.\frac{\partial f}{\partial x}\right|_0 \\
L &= \left.\frac{\partial f}{\partial w}\right|_0 \\
C &= \left.\frac{\partial h}{\partial x}\right|_0 \\
M &= \left.\frac{\partial h}{\partial v}\right|_0
\end{aligned}
\tag{40}
$$

- Compute the following matrices:

$$
\begin{aligned}
\tilde{Q} &= LQL^T \\
\tilde{R} &= MRM^T
\end{aligned}
\tag{41}
$$

- Define Δy as the difference between the actual measurement y and the nominal measurement y_0 :

$$
\Delta y = y - y_0 \tag{42}
$$

- Execute the following Kalman filter equations:

$$
\begin{aligned}
\Delta\hat{x}(0) &= 0 \\
P(0) &= E\left[(\Delta x(0) - \Delta\hat{x}(0))(\Delta x(0) - \Delta\hat{x}(0))^T\right] \\
\Delta\dot{\hat{x}} &= A\Delta\hat{x} + K(\Delta y - C\Delta\hat{x}) \\
K &= PC^T\tilde{R}^{-1} \\
\dot{P} &= AP + PA^T + \tilde{Q} - PC^T\tilde{R}^{-1}CP
\end{aligned}
\tag{43}
$$

- Estimate the state as follows:

$$
\hat{x} = x_0 + \Delta\hat{x} \tag{44}
$$

6.2 Particle Filter

- **Model**:

$$
\begin{aligned}
x_{k+1} &= f_k\left(x_k, w_k\right) \\
y_k &= h_k\left(x_k, v_k\right)
\end{aligned}
\tag{45}
$$

The functions $f_k(\cdot)$ and $h_k(\cdot)$ are time-varying nonlinear system and measurement equations. The noise sequences $\{w_k\}$ and $\{v_k\}$ are assumed to be independent and white with known pdf's.The goal of a Bayesian estimator is to approximate the conditional pdf of x_k based on measurements $y_1, y_2, \cdots, y_k$. This conditional pdf is denoted as

$$
p\left(x_k \mid Y_k\right) = \text{ pdf of } x_k \text{ conditioned on measurements } y_1, y_2, \cdots, y_k \tag{46}
$$

Our goal is to find a recursive way to compute the conditional pdf $p\left(x_k \mid Y_k\right)$ Before we find this conditional pdf, we will find the conditional pdf $p\left(x_k \mid Y_{k-1}\right)$.

$$
\begin{aligned}
p\left(x_k|Y_{k-1}\right) &= \int p\left[\left(x_k, x_{k-1}\right)|Y_{k-1}\right] dx_{k-1} \\
&= \int p\left[x_k|\left(x_{k-1}, Y_{k-1}\right)\right] p\left(x_{k-1}|Y_{k-1}\right) dx_{k-1} \\
&= \int p\left(x_k|x_{k-1}\right) p\left(x_{k-1}|Y_{k-1}\right) dx_{k-1}
\end{aligned}
\tag{47}
$$

Now consider the a posteriori conditional pdf of x_k.

$$
\begin{aligned}
p\left(x_k|Y_k\right) &= \frac{p\left(Y_k|x_k\right)}{p\left(Y_k\right)} p\left(x_k\right) \\
&= \frac{p\left[\left(y_k, Y_{k-1}\right)|x_k\right]}{p\left(y_k, Y_{k-1}\right)} \underbrace{\frac{p\left(x_k|Y_{k-1}\right) p\left(Y_{k-1}\right)}{p\left(Y_{k-1}|x_k\right)}}_{p\left(x_k\right)} \\
&= \frac{p\left(x_k, y_k, Y_{k-1}\right)}{p\left(x_k\right) p\left(y_k, Y_{k-1}\right)} \frac{p\left(x_k, Y_{k-1}\right) p\left(Y_{k-1}\right)}{p\left(Y_{k-1}\right) p\left(Y_{k-1}|x_k\right)} \\
&= \frac{p\left(y_k|x_k\right) p\left(x_k|Y_{k-1}\right)}{p\left(y_k|Y_{k-1}\right)}
\end{aligned}
\tag{48}
$$

- **The algorithm of particle filter**:
  1. Assuming that the pdf of the initial state $p\left(x_0\right)$ is known, randomly generate N initial particles on the basis of the pdf $p\left(x_0\right)$. These particles are denoted $x_{0,\imath}^{+}(i=1, \cdots, N)$. The parameter N is chosen by the user as a trade-off between computational effort and estimation accuracy.

  2. For $k=1,2, \cdots$, do the following. (a) Perform the time propagation step to obtain a priori particles $x_{k,2}^{-}$ using the known process equation and the known pdf of the process noise:

  $$x_{k,\imath}^{-}=f_{k-1}\left(x_{k-1,i}^{+}, w_{k-1}^{2}\right) \quad (i=1, \cdots, N)$$

  where each w_{k-1}^{i} noise vector is randomly generated on the basis of the known pdf of w_{k-1}
  (b) Compute the relative likelihood q_i of each particle $x_{k,\imath}^{-}$ conditioned on the measurement y_k. This is done by evaluating the pdf $p\left(y_k \mid x_{k,\imath}^{-}\right)$ on the basis of the nonlinear measurement equation and the pdf of the measurement noise.
  (c) Scale the relative likelihoods obtained in the previous step as follows:

  $$q_i=\frac{q_i}{\sum_{j=1}^{N} q_j}$$

  Now the sum of all the likelihoods is equal to one.
  (d) Generate a set of a posteriori particles $x_{k,i}^{+}$ on the basis of the relative likelihoods q_i.
  (e) Now that we have a set of particles $x_{k,i}^{+}$ that are distributed according to the pdf $p\left(x_k \mid y_k\right)$, we can compute any desired statistical measure of this pdf. We typically are most interested in computing the mean and the covariance.