『非线性滤波算法』

Dr. Yuan-Li Cai

Spring 2024

0. Outline

我们在实际应用中遇到的系统绝大多数都是非线性的，此时的状态估计远比线性系统要复杂和困难。和线性系统类似，非线性系统的状态估计也可以分为预测、滤波和平滑三类。

这里我们仅讨论若干非线性系统的滤波算法，可以类似地建立对应的预测和平滑算法。

1. 非线性贝叶斯滤波理论

我们首先讨论非线性贝叶斯滤波理论框架，这是所有非线性系统估计理论的基础，具有重要的理论价值和意义。

1.1 问题描述

考虑如下形式的非线性系统:

$$\boldsymbol{x}_{k+1} = \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k, k) + \boldsymbol{w}_k \tag{1}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{h}(\boldsymbol{x}_{k+1}, \boldsymbol{u}_{k+1}, k+1) + \boldsymbol{v}_{k+1} \tag{2}$$

这里考虑的是加性噪声。

假设

1. $\{\boldsymbol{u}_k\}$ 是确定性的输入;
2. $\{\boldsymbol{w}_k\}$、$\{\boldsymbol{v}_k\}$ 和 $\boldsymbol{x}_0$ 相互独立, $\{\boldsymbol{w}_k\}$ 和 $\{\boldsymbol{v}_k\}$ 都是白噪声序列;

3. $\boldsymbol{w}_k$ 的概率分布密度函数为 $f_w(\boldsymbol{w}_k)$；

4. $\boldsymbol{v}_k$ 的概率分布密度函数为 $f_v(\boldsymbol{v}_k)$；

5. $\boldsymbol{x}_0$ 的概率分布密度函数为 $f_x(\boldsymbol{x}_0)$；

贝叶斯滤波就是在给定量测 $\boldsymbol{Y}_1^k = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_k\}$ 下，根据 $f_x(\boldsymbol{x}_0)$ 求 $\boldsymbol{x}_k$ 的验后概率分布密度函数 $f_x(\boldsymbol{x}_k|\boldsymbol{Y}_1^k)$。

1.2 切普曼—郭尔莫洛夫方程

设 $\{\boldsymbol{x}(t)\}$ 是一随机过程，对于任意的正整数 m，如果随机向量

$$[\boldsymbol{x}(t_1), \boldsymbol{x}(t_2), \cdots, \boldsymbol{x}(t_m)]^T = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m]^T$$

的条件概率密度满足

$$f(\boldsymbol{x}_m|\boldsymbol{x}_{m-1}, \boldsymbol{x}_{m-2}, \cdots, \boldsymbol{x}_1) = f(\boldsymbol{x}_m|\boldsymbol{x}_{m-1}) \tag{3}$$

那么称 $\{\boldsymbol{x}(t)\}$ 为一阶马尔科夫过程。其中 $t_1 \sim t_m$ 是任意选取的，$f(\boldsymbol{x}_m|\boldsymbol{x}_{m-1})$ 称为一步转移概率密度函数，$f(\boldsymbol{x}_m|\boldsymbol{x}_{m-l})$ 称为 l 步转移概率密度函数。

对于马尔科夫过程，根据贝叶斯法则易得

$$
\begin{aligned}
& f(\boldsymbol{x}_m, \boldsymbol{x}_{m-1}, \boldsymbol{x}_{m-2}, \cdots, \boldsymbol{x}_1) \\
&= f(\boldsymbol{x}_m | \boldsymbol{x}_{m-1}, \boldsymbol{x}_{m-2}, \cdots, \boldsymbol{x}_1) f(\boldsymbol{x}_{m-1}, \boldsymbol{x}_{m-2}, \cdots, \boldsymbol{x}_1) \\
&= f(\boldsymbol{x}_m | \boldsymbol{x}_{m-1}) f(\boldsymbol{x}_{m-1} | \boldsymbol{x}_{m-2}) \cdots f(\boldsymbol{x}_2 | \boldsymbol{x}_1) f(\boldsymbol{x}_1) \\
&= f(\boldsymbol{x}_1) \prod_{k=1}^{m-1} f(\boldsymbol{x}_{k+1} | \boldsymbol{x}_k)
\end{aligned} \tag{4}
$$

另外，由边缘密度函数计算可知

$$
\int_{-\infty}^{+\infty} f(\boldsymbol{x}_m, \boldsymbol{x}_{m-1}, \boldsymbol{x}_{m-2}) \mathrm{d}\boldsymbol{x}_{m-1} = f(\boldsymbol{x}_m, \boldsymbol{x}_{m-2}) \tag{5}
$$

即

$$\begin{aligned}
&\int_{-\infty}^{+\infty} f(\boldsymbol{x}_m|\boldsymbol{x}_{m-1})f(\boldsymbol{x}_{m-1}|\boldsymbol{x}_{m-2})f(\boldsymbol{x}_{m-2})\mathrm{d}\boldsymbol{x}_{m-1} \\
&\qquad = f(\boldsymbol{x}_m|\boldsymbol{x}_{m-2})f(\boldsymbol{x}_{m-2}) \qquad (6)
\end{aligned}$$

所以

$$f(\boldsymbol{x}_m|\boldsymbol{x}_{m-2}) = \int_{-\infty}^{+\infty} f(\boldsymbol{x}_m|\boldsymbol{x}_{m-1})f(\boldsymbol{x}_{m-1}|\boldsymbol{x}_{m-2})\mathrm{d}\boldsymbol{x}_{m-1} \qquad (7)$$

这就是著名的切普曼—郭尔莫洛夫方程。

更一般地，可以表示为

$$f(\boldsymbol{x}_m|\boldsymbol{x}_{m-l}) = \int_{-\infty}^{+\infty} f(\boldsymbol{x}_m|\boldsymbol{x}_{m-1})f(\boldsymbol{x}_{m-1}|\boldsymbol{x}_{m-l})\mathrm{d}\boldsymbol{x}_{m-1} \tag{8}$$

切普曼—郭尔莫洛夫方程给出了求一步或多步转移概率函数的方法。

1.3 贝叶斯滤波公式

根据贝叶斯法则，我们有

$$
\begin{aligned}
f_x(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^{k+1}) &= f_x(\boldsymbol{x}_{k+1}|\boldsymbol{y}_{k+1}, \boldsymbol{Y}_1^k) \\
&= \frac{f_{xy}(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}, \boldsymbol{Y}_1^k)}{f_y(\boldsymbol{y}_{k+1}, \boldsymbol{Y}_1^k)} \\
&= \frac{f_{xy}(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}|\boldsymbol{Y}_1^k)}{f_y(\boldsymbol{y}_{k+1}|\boldsymbol{Y}_1^k)}
\end{aligned} \tag{9}
$$

注意到（因为 $\boldsymbol{v}_k$ 的统计性质）

$$
\begin{aligned}
f_{xy}(\boldsymbol{x}_{k+1}, \boldsymbol{y}_{k+1}|\boldsymbol{Y}_1^k) &= f_x(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^k) f_y(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1}, \boldsymbol{Y}_1^k) \\
&= f_x(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^k) f_y(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1})
\end{aligned}
$$

(9) 即为

$$f_x(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^{k+1}) = \frac{f_x(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^k)f_y(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1})}{f_y(\boldsymbol{y}_{k+1}|\boldsymbol{Y}_1^k)} \tag{10}$$

根据切普曼—郭尔莫洛夫方程，则有

$$\begin{aligned} f_x(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^k) &= \int_x f_x(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k, \boldsymbol{Y}_1^k)f_x(\boldsymbol{x}_k|\boldsymbol{Y}_1^k)\mathrm{d}\boldsymbol{x}_k \\ &= \int_x f_x(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k)f_x(\boldsymbol{x}_k|\boldsymbol{Y}_1^k)\mathrm{d}\boldsymbol{x}_k \end{aligned} \tag{11}$$

$$\begin{aligned} f_y(\boldsymbol{y}_{k+1}|\boldsymbol{Y}_1^k) &= \int_x f_y(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1},\boldsymbol{Y}_1^k) f_x(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^k)\mathrm{d}\boldsymbol{x}_{k+1} \\ &= \int_x f_y(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1}) f_x(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^k)\mathrm{d}\boldsymbol{x}_{k+1} \end{aligned} \tag{12}$$

而由系统方程 (1)、(2)，可知

$$f_x(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k) = f_w(\boldsymbol{x}_{k+1} - \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k, k)) \tag{13}$$

$$f_y(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1}) = f_v(\boldsymbol{y}_{k+1} - \boldsymbol{h}(\boldsymbol{x}_{k+1}, \boldsymbol{u}_{k+1}, k+1)) \tag{14}$$

将以上二式代入 (11)、(12)，得

$$f_x(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^k) = \int_x f_w(\boldsymbol{x}_{k+1} - \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k, k)) f_x(\boldsymbol{x}_k|\boldsymbol{Y}_1^k)\mathrm{d}\boldsymbol{x}_k \tag{15}$$

$$f_y(\boldsymbol{y}_{k+1}|\boldsymbol{Y}_1^k) = \int_x f_v(\boldsymbol{y}_{k+1} - \boldsymbol{h}(\boldsymbol{x}_{k+1}, \boldsymbol{u}_{k+1}, k+1)) f_x(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^k)\mathrm{d}\boldsymbol{x}_{k+1} \tag{16}$$

在起始时刻

$$f_x(\boldsymbol{x}_0|\boldsymbol{Y}_1^0) = f_x(\boldsymbol{x}_0) \tag{17}$$

(9)、(15) $\sim$ (17) 便构成了计算的递推算式，即贝叶斯滤波算法，见表1。

Table 1: 贝叶斯滤波算法

状态方程	$\boldsymbol{x}_{k+1} = \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k, k) + \boldsymbol{w}_k$
量测方程	$\boldsymbol{y}_{k+1} = \boldsymbol{h}(\boldsymbol{x}_{k+1}, \boldsymbol{u}_{k+1}, k+1) + \boldsymbol{v}_{k+1}$
状态初始概率密度	$f_x(\boldsymbol{x}_0 \vert \boldsymbol{Y}_1^0) = f_x(\boldsymbol{x}_0)$
状态预测概率密度	$f_x(\boldsymbol{x}_{k+1} \vert \boldsymbol{Y}_1^k) = \int_x f_w(\boldsymbol{x}_{k+1} - \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k, k)) f_x(\boldsymbol{x}_k \vert \boldsymbol{Y}_1^k) \mathrm{d}\boldsymbol{x}_k$
量测预测概率密度	$f_y(\boldsymbol{y}_{k+1} \vert \boldsymbol{Y}_1^k) = \int_x f_v(\boldsymbol{y}_{k+1} - \boldsymbol{h}(\boldsymbol{x}_{k+1}, \boldsymbol{u}_{k+1}, k+1)) f_x(\boldsymbol{x}_{k+1} \vert \boldsymbol{Y}_1^k) \mathrm{d}\boldsymbol{x}_{k+1}$
状态验后概率密度	$f_x(\boldsymbol{x}_{k+1} \vert \boldsymbol{Y}_1^{k+1}) = \dfrac{f_x(\boldsymbol{x}_{k+1} \vert \boldsymbol{Y}_1^k) f_y(\boldsymbol{y}_{k+1} \vert \boldsymbol{x}_{k+1})}{f_y(\boldsymbol{y}_{k+1} \vert \boldsymbol{Y}_1^k)}$

有了状态的验后概率密度，可以根据需要，进一步求出状态的滤波值及误差协方差等低阶统计信息。例如

$$\hat{\boldsymbol{x}}_{k+1|k+1} = E(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^{k+1}) = \int_x \boldsymbol{x}_{k+1} f_x(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^{k+1})\mathrm{d}\boldsymbol{x}_{k+1} \tag{18}$$

$$P_{k+1|k+1} = E(\tilde{\boldsymbol{x}}_{k+1}\tilde{\boldsymbol{x}}_{k+1}^T) = \int_x \tilde{\boldsymbol{x}}_{k+1}\tilde{\boldsymbol{x}}_{k+1}^T f_x(\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^{k+1})\mathrm{d}\boldsymbol{x}_{k+1} \tag{19}$$

对于线性系统，上述公式可以解析求解，即可导出卡尔曼滤波算法。一般情况下，需要求高维的非线性积分，通常非常困难，由此发展出了许多近似处理方法。

2. 基于标称状态的线性化滤波方法

非线性系统线性化是解决工程问题常用而且有效的方法。如果系统的真实状态 x 总是围绕标称状态 $\boldsymbol{x}^*$ 附近变化，当标称状态 $\boldsymbol{x}^*$ 已知时，我们可以取 $x = \boldsymbol{x}^* + \Delta x$，从而建立一种可行的滤波算法。

考虑如下非线性离散时间随机系统

$$\boldsymbol{x}_{k+1} = \boldsymbol{g}(\boldsymbol{x}_k, k) + \boldsymbol{w}_k \tag{20}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{h}(\boldsymbol{x}_{k+1}, k+1) + \boldsymbol{v}_{k+1} \tag{21}$$

其中，过程噪声 $\boldsymbol{w}_k \sim (0, Q_k)$ 与量测噪声 $\boldsymbol{v}_k \sim (0, R_k)$ 互不相关，设 $Q_k \geq 0, R_k > 0$。

标称状态及量测方程定义为

$$\boldsymbol{x}_{k+1}^* = \boldsymbol{g}(\boldsymbol{x}_k^*, k) \tag{22}$$

$$\boldsymbol{y}_{k+1}^* = \boldsymbol{h}(\boldsymbol{x}_{k+1}^*, k+1) \tag{23}$$

设

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_{k+1}^* + \Delta \boldsymbol{x}_{k+1}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{y}_{k+1}^* + \Delta \boldsymbol{y}_{k+1}$$

一阶近似地

$$\Delta \boldsymbol{x}_{k+1} = \Phi_{k+1,k} \Delta \boldsymbol{x}_k + \boldsymbol{w}_k \tag{24}$$

$$\Delta \boldsymbol{y}_{k+1} = H_{k+1} \Delta \boldsymbol{x}_{k+1} + \boldsymbol{v}_{k+1} \tag{25}$$

其中

$$\Phi_{k+1,k} = \left. \frac{\partial \boldsymbol{g}(\boldsymbol{x}_k, k)}{\partial \boldsymbol{x}_k^{\mathrm{T}}} \right|_{\boldsymbol{x}_k = \boldsymbol{x}_k^*}, \quad H_k = \left. \frac{\partial \boldsymbol{h}(\boldsymbol{x}_k, k)}{\partial \boldsymbol{x}_k^{\mathrm{T}}} \right|_{\boldsymbol{x}_k = \boldsymbol{x}_k^*}$$

如果 $\boldsymbol{x}_0 \sim (\boldsymbol{x}_0^*, P_0)$，即 $\Delta \boldsymbol{x}_0 \sim (0, P_0)$，而且与过程噪声 $\boldsymbol{w}_k \sim (0, Q_k)$ 与量测噪声 $\boldsymbol{v}_k \sim (0, R_k)$ 互不相关。那么，我们可以应用卡尔曼滤波算法于 (24)、(25)，从而得到原系统一种滤波算法，如表2所示。

Table 2: 基于标称状态线性化的卡尔曼滤波算法

状态方程	$\boldsymbol{x}_{k+1} = \boldsymbol{g}(\boldsymbol{x}_k, k) + \boldsymbol{w}_k$
量测方程	$\boldsymbol{y}_{k+1} = \boldsymbol{h}(\boldsymbol{x}_{k+1}, k+1) + \boldsymbol{v}_{k+1}$
滤波初值	$\hat{\boldsymbol{x}}_{0\|0} = E\boldsymbol{x}_0 = \boldsymbol{x}_0^*, P_{0\|0} = \mathrm{var}(\boldsymbol{x}_0) = P_0$
一步预测	$\hat{\boldsymbol{x}}_{k+1\|k} = \boldsymbol{g}(\hat{\boldsymbol{x}}_{k\|k}, k)$ $P_{k+1\|k} = \Phi_{k+1,k} P_{k\|k} \Phi_{k+1,k}^T + Q_k$

滤波增益	$K_{k+1} = P_{k+1\|k} H_{k+1}^T (H_{k+1} P_{k+1\|k} H_{k+1}^T + R_{k+1})^{-1}$
滤波计算	$\hat{\boldsymbol{x}}_{k+1\|k+1} = \hat{\boldsymbol{x}}_{k+1\|k} + K_{k+1}[\boldsymbol{y}_{k+1} - \boldsymbol{h}(\hat{\boldsymbol{x}}_{k+1\|k}, k+1)]$ $P_{k+1\|k+1} = (I - K_{k+1} H_{k+1}) P_{k+1\|k}$
辅助方程	$\Phi_{k+1,k} = \left.\dfrac{\partial \boldsymbol{g}(\boldsymbol{x}_k, k)}{\partial \boldsymbol{x}_k^{\mathrm{T}}}\right\|_{\boldsymbol{x}_k = \boldsymbol{x}_k^*}$ $H_k = \left.\dfrac{\partial \boldsymbol{h}(\boldsymbol{x}_k, k)}{\partial \boldsymbol{x}_k^{\mathrm{T}}}\right\|_{\boldsymbol{x}_k = \boldsymbol{x}_k^*}$

[Remarks]

1. 一步预测

$$\hat{\boldsymbol{x}}_{k+1|k} = E[\boldsymbol{x}_{k+1}|\boldsymbol{Y}_1^k]$$

$$\begin{aligned}
\hat{\boldsymbol{x}}_{k+1|k} &= E[\boldsymbol{g}(\boldsymbol{x}_k, k) + \boldsymbol{w}_k|\boldsymbol{Y}_1^k] = E[\boldsymbol{g}(\boldsymbol{x}_k, k)|\boldsymbol{Y}_1^k] \\
&\simeq E[\boldsymbol{g}(\boldsymbol{x}_k^*, k) + \Phi_{k+1,k}(\boldsymbol{x}_k^* - \boldsymbol{x}_k)|\boldsymbol{Y}_1^k] \\
&= \underbrace{\boldsymbol{g}(\boldsymbol{x}_k^*, k) + \Phi_{k+1,k}(\boldsymbol{x}_k^* - \hat{\boldsymbol{x}}_{k|k})}_{\text{由此计算}P_{k+1|k}} \\
&\simeq \boldsymbol{g}(\hat{\boldsymbol{x}}_{k|k}, k)
\end{aligned}$$

2. 量测预报

$$\hat{\boldsymbol{y}}_{k+1|k} = E[\boldsymbol{y}_{k+1}|\boldsymbol{Y}_1^k]$$

$$\begin{aligned}
\hat{\boldsymbol{y}}_{k+1|k} &= E[\boldsymbol{h}(\boldsymbol{x}_{k+1}, k+1) + \boldsymbol{v}_{k+1}|\boldsymbol{Y}_1^k] \\
&= E[\boldsymbol{h}(\boldsymbol{x}_{k+1}, k+1)|\boldsymbol{Y}_1^k] \\
&\simeq E[\boldsymbol{h}(\boldsymbol{x}^*_{k+1}, k+1) + H_{k+1}(\boldsymbol{x}^*_{k+1} - \boldsymbol{x}_{k+1})|\boldsymbol{Y}_1^k] \\
&= \boldsymbol{h}(\boldsymbol{x}^*_{k+1}, k+1) + H_{k+1}(\boldsymbol{x}^*_{k+1} - \hat{\boldsymbol{x}}_{k+1|k}) \\
&\simeq \boldsymbol{h}(\hat{\boldsymbol{x}}_{k+1|k}, k+1)
\end{aligned}$$

3. 扩展卡尔曼滤波

研究如下非线性离散时间随机系统:

$$\boldsymbol{x}_{k+1} = \boldsymbol{g}(\boldsymbol{x}_k, k) + \boldsymbol{w}_k \tag{26}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{h}(\boldsymbol{x}_{k+1}, k+1) + \boldsymbol{v}_{k+1} \tag{27}$$

其中，过程噪声 $\boldsymbol{w}_k \sim (0, Q_k)$ 与量测噪声 $\boldsymbol{v}_k \sim (0, R_k)$ 互不相关，设 $Q_k \geq 0, R_k > 0$； $\boldsymbol{x}_0 \sim (\bar{\boldsymbol{x}}_0, P_0)$ 与 $\boldsymbol{w}_k \sim (0, Q_k)$、$\boldsymbol{v}_k \sim (0, R_k)$ 均不相关。

定义

$$\Phi_{k+1,k} = \left.\frac{\partial \boldsymbol{g}(\boldsymbol{x}_k, k)}{\partial \boldsymbol{x}_k{}^{\mathrm{T}}}\right|_{\boldsymbol{x}_k=\hat{\boldsymbol{x}}_{k|k}} \tag{28}$$

$$H_{k+1} = \left.\frac{\partial \boldsymbol{h}(\boldsymbol{x}_{k+1}, k+1)}{\partial \boldsymbol{x}_{k+1}^{\mathrm{T}}}\right|_{\boldsymbol{x}_{k+1}=\hat{\boldsymbol{x}}_{k+1|k}} \tag{29}$$

那么

$$\boldsymbol{x}_{k+1} \approx \Phi_{k+1,k}\boldsymbol{x}_k + \left[\boldsymbol{g}(\hat{\boldsymbol{x}}_{k|k}, k) - \Phi_{k+1,k}\hat{\boldsymbol{x}}_{k|k}\right] + \boldsymbol{w}_k \tag{30}$$

$$\boldsymbol{y}_{k+1} \approx H_{k+1}\boldsymbol{x}_{k+1} + \left[\boldsymbol{h}(\hat{\boldsymbol{x}}_{k+1|k}, k+1) - H_{k+1}\hat{\boldsymbol{x}}_{k+1|k}\right] + \boldsymbol{v}_{k+1} \tag{31}$$

因为在求 $k+1$ 时刻状态的滤波时，$\hat{\boldsymbol{x}}_{k|k}$、$\hat{\boldsymbol{x}}_{k+1|k}$ 已经知道，可以视为确定性信号。即可以将 $\boldsymbol{g}(\hat{\boldsymbol{x}}_{k|k}, k) - \Phi_{k+1,k}\hat{\boldsymbol{x}}_{k|k}$ 与 $\boldsymbol{h}(\hat{\boldsymbol{x}}_{k+1|k}, k+1) - H_{k+1}\hat{\boldsymbol{x}}_{k+1|k}$

视为确定性输入信号。这样，我们便可以借助卡尔曼滤波基本方程建立一套有效的非线性滤波算法，称为扩展卡尔曼滤波 (EKF)，如表3所示。

Table 3: 扩展卡尔曼滤波算法

状态方程	$\boldsymbol{x}_{k+1} = \boldsymbol{g}(\boldsymbol{x}_k, k) + \boldsymbol{w}_k$
量测方程	$\boldsymbol{y}_{k+1} = \boldsymbol{h}(\boldsymbol{x}_{k+1}, k+1) + \boldsymbol{v}_{k+1}$
滤波初值	$\hat{\boldsymbol{x}}_{0\|0} = E\boldsymbol{x}_0 = \bar{\boldsymbol{x}}_0, P_{0\|0} = \mathrm{var}(\boldsymbol{x}_0) = P_0$

一步预测	$\hat{\boldsymbol{x}}_{k+1\|k} = \boldsymbol{g}(\hat{\boldsymbol{x}}_{k\|k}, k)$ $P_{k+1\|k} = \Phi_{k+1,k} P_{k\|k} \Phi_{k+1,k}^T + Q_k$
滤波增益	$K_{k+1} = P_{k+1\|k} H_{k+1}^T (H_{k+1} P_{k+1\|k} H_{k+1}^T + R_{k+1})^{-1}$
滤波计算	$\hat{\boldsymbol{x}}_{k+1\|k+1} = \hat{\boldsymbol{x}}_{k+1\|k} + K_{k+1}[\boldsymbol{y}_{k+1} - \boldsymbol{h}(\hat{\boldsymbol{x}}_{k+1\|k}, k+1)]$ $P_{k+1\|k+1} = (I - K_{k+1} H_{k+1}) P_{k+1\|k}$

辅助方程

$$\Phi_{k+1,k} = \left.\frac{\partial \boldsymbol{g}(\boldsymbol{x}_k, k)}{\partial \boldsymbol{x}_k^{\mathrm{T}}}\right|_{\boldsymbol{x}_k=\hat{\boldsymbol{x}}_{k|k}}$$

$$H_{k+1} = \left.\frac{\partial \boldsymbol{h}(\boldsymbol{x}_{k+1}, k+1)}{\partial \boldsymbol{x}_{k+1}^{\mathrm{T}}}\right|_{\boldsymbol{x}_{k+1}=\hat{\boldsymbol{x}}_{k+1|k}}$$

上述扩展卡尔曼滤波算法是解决非线性滤波使用最广泛的方法，大量实际应用证明是行之有效的。

[Remarks]

1. 一步预测

$$\hat{\boldsymbol{x}}_{k+1|k} = E[\boldsymbol{x}_{k+1}|\mathbf{Y}_1^k]$$

$$\begin{aligned}\hat{\boldsymbol{x}}_{k+1|k} &= E[\boldsymbol{g}(\boldsymbol{x}_k, k) + \boldsymbol{w}_k|\mathbf{Y}_1^k] = E[\boldsymbol{g}(\boldsymbol{x}_k, k)|\mathbf{Y}_1^k] \\ &\simeq E[\boldsymbol{g}(\hat{\boldsymbol{x}}_{k|k}, k) + \Phi_{k+1,k}(\hat{\boldsymbol{x}}_{k|k} - \boldsymbol{x}_k)|\mathbf{Y}_1^k] \\ &= \boldsymbol{g}(\hat{\boldsymbol{x}}_{k|k}, k)\end{aligned}$$

2. 量测预报

$$\hat{\boldsymbol{y}}_{k+1|k} = E[\boldsymbol{y}_{k+1}|\mathbf{Y}_1^k]$$

$$
\begin{aligned}
\hat{\boldsymbol{y}}_{k+1|k} &= E[\boldsymbol{h}(\boldsymbol{x}_{k+1}, k+1) + \boldsymbol{v}_{k+1} | \boldsymbol{Y}_1^k] \\
&= E[\boldsymbol{h}(\boldsymbol{x}_{k+1}, k+1) | \boldsymbol{Y}_1^k] \\
&\simeq E[\boldsymbol{h}(\hat{\boldsymbol{x}}_{k+1|k}, k+1) + H_{k+1}(\hat{\boldsymbol{x}}_{k+1|k} - \boldsymbol{x}_{k+1}) | \boldsymbol{Y}_1^k] \\
&= \boldsymbol{h}(\hat{\boldsymbol{x}}_{k+1|k}, k+1)
\end{aligned}
$$

4. 递代扩展卡尔曼滤波

设按上一节介绍的扩展卡尔曼滤波器已经获得了 $\hat{\boldsymbol{x}}_{k|k}$、$\hat{\boldsymbol{x}}_{k|k-1}$，记

$$\hat{\boldsymbol{x}}_{k|k}^{(0)} = \hat{\boldsymbol{x}}_{k|k-1}$$

可以以此进一步改善 $\hat{\boldsymbol{y}}_{k|k-1}$ 的计算。

定义

$$H(\hat{\boldsymbol{x}}_{k|k}^{(i)}, k) = \left.\frac{\partial \boldsymbol{h}(\boldsymbol{x}_k, k)}{\partial \boldsymbol{x}_k^{\mathrm{T}}}\right|_{\boldsymbol{x}_k=\hat{\boldsymbol{x}}_{k|k}^{(i)}}, \quad i = 0, 1, 2, \cdots$$

可得

$$\boldsymbol{y}_k \approx H(\hat{\boldsymbol{x}}_k^{(i)}, k)\boldsymbol{x}_k + \boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k}^{(i)}, k) - H(\hat{\boldsymbol{x}}_k^{(i)}, k)\hat{\boldsymbol{x}}_{k|k}^{(i)} + \boldsymbol{v}_k$$

于是

$$\begin{aligned}\hat{\boldsymbol{y}}_{k|k-1}^{(i)} &\approx \boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k}^{(i)}, k) + H(\hat{\boldsymbol{x}}_k^{(i)}, k)(\hat{\boldsymbol{x}}_{k|k-1} - \hat{\boldsymbol{x}}_{k|k}^{(i)}) \\ &\approx \boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k}^{(i)}, k)\end{aligned}$$

从而可得 $\hat{\boldsymbol{x}}_{k|k}$ 更好的估计 $\hat{\boldsymbol{x}}_{k|k}^{(i+1)}$，即

$$\hat{\boldsymbol{x}}_{k|k}^{(i+1)} = \hat{\boldsymbol{x}}_{k|k-1} + K_k^{(i)}(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|k-1}^{(i)}) \tag{32}$$

$$K_k^{(i)} = P_{k|k-1}H^T(\hat{\boldsymbol{x}}_{k|k}^{(i)}, k)(H(\hat{\boldsymbol{x}}_{k|k}^{(i)}, k)P_{k|k-1}H^T(\hat{\boldsymbol{x}}_{k|k}^{(i)}, k) + R_{k+1})^{-1} \tag{33}$$

$$P_{k|k}^{(i+1)} = (I - K_k^{(i)}H(\hat{\boldsymbol{x}}_{k|k}^{(i)}, k))P_{k+1|k} \tag{34}$$

按上述方程反复递代（i=1,2,3....），直到不能改善为止。这便是 递代扩展卡尔曼滤波 (IEKF)，其精度将高于普通扩展卡尔曼滤波，但计算时间会增加。以计算时间增加换取滤波精度提高。工程应用中，为了保证滤波算法的实时性，迭代次数一般取 $3 \sim 5$。

5. Unscented Kalman Filter(UKF)

UKF Link

6. 粒子滤波 (PF)

PF LINK

□

Questions?