

杨亮《从贝叶斯开始学滤波》中一些问题

A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation (Anastasios I. Mourikis and Stergios I. Roumeliotis)

这篇文章的主要思想就是用 IMU 和一个单目相机，单目相机的更新频率没有 IMU 的频率高。两个数据，要考虑两种数据的时间戳能否对上，在这种情况下，采用融合的方式。一个数据用来预测，另一个数据用来观测，然后用协方差来转换权重。

为什么要讲随机状态和估计？

因为这是一个概率问题。

预测的误差如何累积？

因为我在一个开环的估计里面，不断地走的时候没有人来告诉你走到哪里了，这个误差就会累积。

融合如何进行？

例如图形和加速度不同的信息如何融合？

贝叶斯准则基本理论

贝叶斯准则的基本概念[1]:

两个随机变量 X 和 Y 的联合分布由下式给出:

$$p(x, y) = p(X = x, Y = y)$$

如果 X 和 Y 相互独立，则有

$$p(x, y) = p(x)p(y)$$

在已经知道 $Y=y$ 的基础上，求 X 的条件概率:

$$p(x|y) = \frac{p(x, y)}{p(y)}$$

如果 X 和 Y 相互独立:

$$p(x|y) = \frac{p(x, y)}{p(y)} = p(x)$$

全概率定理:

$$p(x) = \sum_y p(x|y)p(y)$$
$$p(x) = \int p(x|y)p(y)dy$$

贝叶斯准则:

$$p(x|y) = \frac{P(y|x)p(x)}{p(y)} = \frac{p(y|x)p(x)}{\sum_{x'} p(y|x')p(x')}$$

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} = \frac{p(y|x)p(x)}{\int p(y|x')p(x')dx'}$$

利用贝叶斯法则，有

$$p(x|z) = \frac{p(z|x)p(x)}{p(z)} \propto p(z|x)p(x)$$

贝叶斯法则左侧通常称为后验概率，右侧 $p(z|x)$ 称为似然，另一部分 $p(x)$ 称为先验。直接求后验分布是困难的，但是求一个状态最优估计，使得在该状态下后验概率最大化 (Maximize a Posterior, MAP)，则是可行的：

$$x_{MAP}^* = \arg \max p(x|z) = \arg \max p(z|x)p(x)$$

贝叶斯法则的分母部分与带估计的状态 x 无关，因而可以忽略。贝叶斯法则告诉我们，求解最大后验概率相当于最大化似然和先验的乘积。进一步，没有先验时，可以求解 x 的最大似然估计 (Maximize Likelihood Estimation, MLE)。

$$x_{MLE}^* = \arg \max p(z|x)$$

似然：在现在的位姿下，可能产生怎样的观测数据。

因为我们知道观测数据，最大似然可以理解为：在什么样的状态下，最可能产生现在观测到的数据。

贝叶斯滤波

在本节的贝叶斯滤波[2]中，公式中的符号表示如下：

($\overset{\wedge}{\bullet}$) 上帽子来表示后验估计

($\overset{\vee}{\bullet}$) 下帽子来表示先验估计

$$p(x_k | \overset{\vee}{x}_0, v_{1:k}, y_{0:k}) = \eta p(y_k | x_k) p(x_k | \overset{\vee}{x}_0, v_{1:k}, y_{0:k-1}) \quad (1.1)$$

式(1.1)根据全概率公理进行推导，对式(1.1)引入隐藏状态 x_{k-1} ，右边部分

$$p(x_k | \overset{\vee}{x}_0, v_{1:k}, y_{0:k-1}) = \int p(x_k, x_{k-1} | \overset{\vee}{x}_0, v_{1:k}, y_{0:k-1}) dx_{k-1} \quad (1.2)$$

$$= \int p(x_k | x_{k-1}, \overset{\vee}{x}_0, v_{1:k}, y_{0:k-1}) p(x_{k-1} | \overset{\vee}{x}_0, v_{1:k}, y_{0:k-1}) dx_{k-1}$$

$$p(x_k | x_{k-1}, \overset{\vee}{x}_0, v_{1:k}, y_{0:k-1}) = p(x_k | x_{k-1}, v_k) \quad (1.3)$$

$$p(x_{k-1} | \overset{\vee}{x}_0, v_{1:k}, y_{0:k-1}) = p(x_{k-1} | \overset{\vee}{x}_0, v_{1:k-1}, y_{0:k-1}) \quad (1.4)$$

$$p(x_k | \overset{\vee}{x}_0, v_{1:k}, y_{0:k}) = \eta p(y_k | x_k) \int p(x_k | x_{k-1}, v_k) p(x_{k-1} | \overset{\vee}{x}_0, v_{1:k-1}, y_{0:k-1}) dx_{k-1} \quad (1.5)$$

$p(y_k | x_k)$ 通过 $g(\bullet)$ 更新, $p(x_k | x_{k-1}, v_k)$ 利用 $f(\bullet)$ 进行预测, $p(x_{k-1} | \overset{\vee}{x}_0, v_{1:k-1}, y_{0:k-1})$ 是先验置信度。

卡尔曼滤波器

预测的高斯分布:

$$y(x, x_p, \delta_p) = \frac{1}{\sqrt{2\pi\delta_p^2}} e^{-\frac{(x-x_p)^2}{2\delta_p^2}}$$

观测的高斯分布:

$$y(x, x_m, \delta_m) = \frac{1}{\sqrt{2\pi\delta_m^2}} e^{-\frac{(x-x_m)^2}{2\delta_m^2}}$$

有观测也有预测:

$$y(x, x_p, \delta_p, x_m, \delta_m) = \frac{1}{2\pi\sqrt{\delta_m^2\delta_p^2}} e^{-\frac{(x-x_p)^2}{2\delta_p^2} - \frac{(x-x_m)^2}{2\delta_m^2}}$$

假设融合之后的为:

$$y(x, x_f, \delta_f) = \frac{1}{\sqrt{2\pi\delta_f^2}} e^{-\frac{(x-x_f)^2}{2\delta_f^2}}$$

求解得到

$$\begin{aligned} \delta_f^2 &= \delta_p^2 - \frac{\delta_p^4}{\delta_p^2 + \delta_m^2} \\ x_f &= x_p + \frac{\delta_p^2(x_m - x_p)}{\delta_p^2 + \delta_m^2} \end{aligned} \quad (1.6)$$

通过上式 (1.6) 可以看出,

当 δ_p 越大时, 说明我的预测值越不准, 所以我更应该去相信测量值 x_m , 因此 x_m 的系数越

接近于 1, x_f 越接近于 x_m 。

当 δ_p 越小时, 说明我的预测值越准, 所以我更应该去相信测量值 x_p , 因此 x_m 的系数越接近

于 0, x_f 越接近于 x_p 。

因为测量值是从观测来的, 所以有:

$$Z = HX, \delta_z = H\delta_m$$

根据式(1.6)，可以得到卡尔曼增益

$$K = H\delta_p^2 / (H^2\delta_p^2 + \delta_m^2) \quad (1.7)$$

式(1.7)在式(1.6)的增益的基础上有修改，主要是分子上的 H 。

从上面可以看出，卡尔曼滤波器就是根据高斯模型，简化了在计算时候的一些问题。然后再通过两个量之间的不确定量（方差）来融合估计和观测。

扩展卡尔曼滤波器和 SLAM

线性：KF

非线性或者部分非线性：EKF

扩展卡尔曼滤波器（EKF）

将置信度和噪声限制为高斯分布，并且对运动模型和观测模型进行线性化，计算贝叶斯滤波中的积分（以及归一化积），得到 EKF。

对于一般性的状态模型，定义研究对象的运动和观测模型为，

$$\begin{aligned} \text{运动方程: } x_k &= f(x_{k-1}, v_k, w_k), k=1, \dots, K \\ \text{观测方程: } y_k &= g(x_k, n_k), k=0, \dots, K \end{aligned} \quad (1.8)$$

我们通过线性化将其恢复为加性噪声（取近似）的形式：

$$\begin{aligned} x_k &= f(x_{k-1}, v_k) + w_k \\ y_k &= g(x_k) + n_k \end{aligned} \quad (1.9)$$

式(1.9)其实是式(1.8)的特殊情况。

由于 $g(\bullet)$ 和 $f(\bullet)$ 的非线性特性，我们无法计算得到贝叶斯滤波器中的积分的解析解，转而使用线性化的方法，在当前均值（后验状态估计）处展开，对运动和观测模型进行线性化。

$$f(x_{k-1}, v_k, w_k) \approx \check{x}_k + F_{k-1}(x_{k-1} - \check{x}_{k-1}) + w_k' \quad (1.10)$$

$$g(x_k, n_k) \approx \check{y}_k + G_{k-1}(x_k - \check{x}_k) + n_k' \quad (1.11)$$

最后推导得到的 EKF 经典递归方程如下：

预测：

$$\begin{aligned} \check{P}_k &= F_{k-1} \hat{P}_{k-1} F_{k-1}^T + Q_k' \\ \check{x}_k &= f(\check{x}_{k-1}, v_k, 0) \end{aligned} \quad (1.12)$$

卡尔曼增益：

$$K_k = \check{P}_k G_k^T (G_k \check{P}_k G_k^T + R_k')^{-1} \quad (1.13)$$

更新：

$$\begin{aligned}\hat{\mathbf{P}}_k &= (1 - \mathbf{K}_k \mathbf{G}_k) \check{\mathbf{P}}_k \\ \hat{\mathbf{x}}_k &= \check{\mathbf{x}}_k + \mathbf{K}_k (\mathbf{y}_k - \mathbf{g}(\check{\mathbf{x}}_k, 0))\end{aligned}\quad (1.14)$$

对以上公式的说明：

$$\mathbf{F}_{k-1} = \left. \frac{\partial \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{v}_k, \mathbf{w}_k)}{\partial \mathbf{x}_{k-1}} \right|_{\hat{\mathbf{x}}_{k-1}, \mathbf{v}_k, 0} \quad (1.15)$$

$$\begin{aligned}\mathbf{Q}_k' : \\ p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{v}_k) &\approx N(\check{\mathbf{x}}_k + \mathbf{F}_{k-1}(\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}), \mathbf{Q}_k') \\ \mathbf{Q}_k' &= E[\mathbf{w}_k' \mathbf{w}_k'^T] \approx E[(\mathbf{x}_k - E[\mathbf{x}_k])(\mathbf{x}_k - E[\mathbf{x}_k])^T]\end{aligned}\quad (1.16)$$

$$\mathbf{G}_k = \left. \frac{\partial \mathbf{g}(\mathbf{x}_k, \mathbf{n}_k)}{\partial \mathbf{x}_k} \right|_{\check{\mathbf{x}}_k, 0} \quad (1.17)$$

$$\mathbf{R}_k' = E[\mathbf{n}_k' \mathbf{n}_k'^T] \approx E[(\mathbf{y}_k - E[\mathbf{y}_k])(\mathbf{y}_k - E[\mathbf{y}_k])^T] \quad (1.18)$$

EKF 的主要问题在于，其线性化的工作点是估计状态的均值，而不是真实状态。这可能导致 EKF 在某些情况下快速地发散。

迭代扩展卡尔曼滤波器（IEKF）

对非线性观测模型

$$\mathbf{y}_k = \mathbf{g}(\mathbf{x}_k, \mathbf{n}_k) \quad (1.19)$$

选取一个点 $\mathbf{x}_{op,k}$ 进行线性化，得到：

$$\begin{aligned}\mathbf{g}(\mathbf{x}_k, \mathbf{n}_k) &\approx \mathbf{y}_{op,k} + \mathbf{G}_k(\mathbf{x}_k - \mathbf{x}_{op,k}) + \mathbf{n}_k' \\ \mathbf{y}_{op,k} &= \mathbf{g}(\mathbf{x}_{op,k}, 0), \mathbf{G}_k = \left. \frac{\partial \mathbf{g}(\mathbf{x}_k, \mathbf{n}_k)}{\partial \mathbf{x}_k} \right|_{\mathbf{x}_{op,k}, 0}, \mathbf{n}_k' = \left. \frac{\partial \mathbf{g}(\mathbf{x}_k, \mathbf{n}_k)}{\partial \mathbf{n}_k} \right|_{\mathbf{x}_{op,k}, 0} \mathbf{n}_k\end{aligned}\quad (1.20)$$

将时刻为 k 处的状态和测量的两盒概率近似为高斯分布：

$$\begin{aligned}p(\mathbf{x}_k, \mathbf{y}_k | \check{\mathbf{x}}_0, \mathbf{v}_{1:k}, \mathbf{y}_{0:k-1}) &\approx N\left(\begin{bmatrix} \mu_{x,k} \\ \mu_{y,k} \end{bmatrix}, \begin{bmatrix} \Sigma_{xx,k} & \Sigma_{xy,k} \\ \Sigma_{yx,k} & \Sigma_{yy,k} \end{bmatrix}\right) \\ &= N\left(\begin{bmatrix} \check{\mathbf{x}}_k \\ \mathbf{y}_{op,k} + \mathbf{G}_k(\check{\mathbf{x}}_k - \mathbf{x}_{op,k}) \end{bmatrix}, \begin{bmatrix} \check{\mathbf{P}}_k & \check{\mathbf{P}}_k \mathbf{G}_k^T \\ \mathbf{G}_k \check{\mathbf{P}}_k^T & \mathbf{G}_k \check{\mathbf{P}}_k \mathbf{G}_k^T + \mathbf{R}_k' \end{bmatrix}\right)\end{aligned}\quad (1.21)$$

然后结合广义高斯滤波的校正方程，得到 IEKF 的公式：

$$\begin{aligned}
\mathbf{K}_k &= \check{\mathbf{P}}_k \mathbf{G}_k^T (\mathbf{G}_k \check{\mathbf{P}}_k \mathbf{G}_k^T + \mathbf{R}_k)^{-1} \\
\hat{\mathbf{P}}_k &= (1 - \mathbf{K}_k \mathbf{G}_k) \check{\mathbf{P}}_k \\
\hat{\mathbf{x}}_k &= \check{\mathbf{x}}_k + \mathbf{K}_k (\mathbf{y}_k - \mathbf{y}_{op,k} - \mathbf{G}_k (\check{\mathbf{x}}_k - \mathbf{x}_{op,k}))
\end{aligned} \tag{1.22}$$

IEKF 与 EKF 的公式的唯一区别在于线性化的工作点，如果我们让线性化的工作点为预测先验的均值，即 $\mathbf{x}_{op,k} = \check{\mathbf{x}}_k$ ，那么 IEKF 就和 EKF 完全相同。

但是，如果我们让每次迭代中的工作点设置为上一次迭代的后验均值，将得到更好的结果。

$$\mathbf{x}_{op,k} \leftarrow \hat{\mathbf{x}}_k \tag{1.23}$$

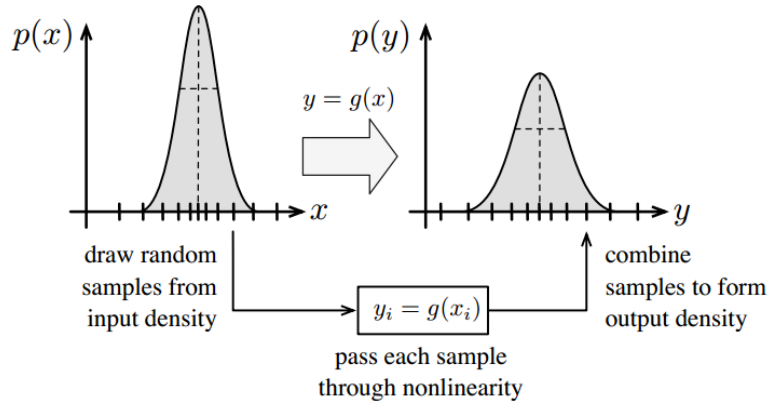
只有在第一次迭代时，令 $\mathbf{x}_{op,k} = \check{\mathbf{x}}_k$ ，其他情况都参照公式(1.23)，进行迭代。

sigmapoint 卡尔曼滤波器/无迹卡尔曼滤波器 (SPKF/UKF)

将概率分布函数传入非线性函数的方法大概有 3 种：蒙特卡罗方法，线性化方法以及 sigmapoint 方法。

1. 蒙特卡罗方法

Figure 4.8 Monte Carlo method to transform a PDF through a nonlinearity. A large number of random samples are drawn from the input density, passed through the nonlinearity, and then used to form the output density.



Monte Carlo Method

2. 线性化方法

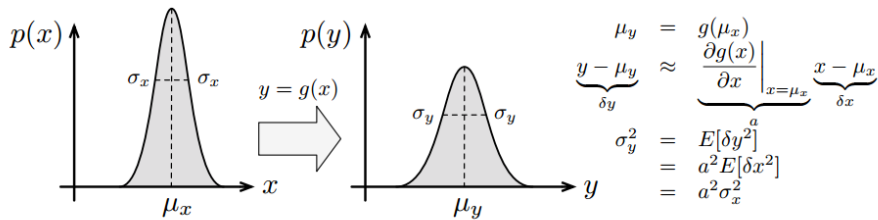


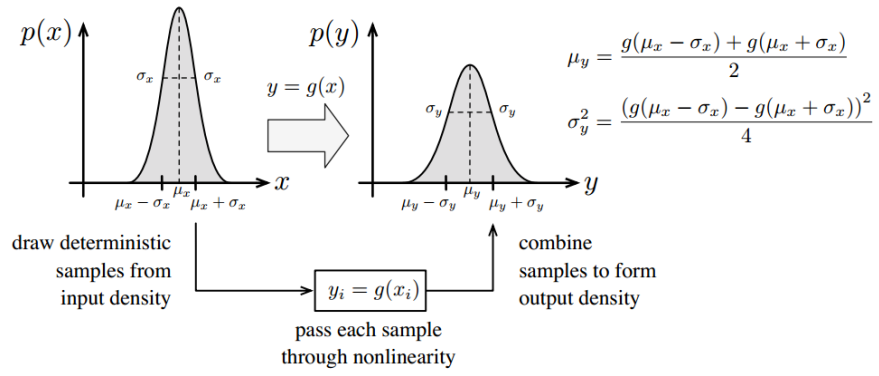
Figure 4.9 One-dimensional Gaussian PDF transformed through a deterministic nonlinear function, $g(\cdot)$. Here we linearize the nonlinearity to propagate the variance approximately.

approach the correct answer with a large number of samples, but as we will see, some of the other methods cannot.

Linearization

3. sigmapoint 方法

Figure 4.10
One-dimensional Gaussian PDF transformed through a deterministic nonlinear function, $g(\cdot)$. Here the basic sigmapoint transformation is used in which only two deterministic samples (one on either side of the mean) approximate the input density.



在非线性的运动和观测模型中，不采用线性化的方法，而是使用 sigmapoint 变换来传递概率分布函数。这样就导出了 sigmapoint 卡尔曼滤波器 (SPKF), 也称为无迹卡尔曼滤波 (UKF)。

SPKF 分为 2 个部分，第一部分为预测步骤，第二部分是校正步骤。

预测：

预测步骤主要内容是将先验置信度 $\{\hat{\mathbf{x}}_{k-1}, \hat{\mathbf{P}}_{k-1}\}$ 转换为预测置信度 $\{\hat{\mathbf{x}}_k, \hat{\mathbf{P}}_k\}$

1. 先验置信度和运动噪声都有不确定性，将他们按以下方式堆叠在一起：

$$\boldsymbol{\mu}_z = \begin{bmatrix} \hat{\mathbf{x}}_{k-1} \\ 0 \end{bmatrix}, \Sigma_{zz} = \begin{bmatrix} \hat{\mathbf{P}}_{k-1} & 0 \\ 0 & \mathbf{Q}_k \end{bmatrix} \quad (1.24)$$

$$L = \dim(\boldsymbol{\mu}_z) \quad (1.25)$$

2. 将 $\{\boldsymbol{\mu}_z, \Sigma_{zz}\}$ 转化为 sigmapoint 表示：

$$\begin{aligned} \mathbf{L}\mathbf{L}^T &= \Sigma_{zz} \\ \mathbf{z}_0 &= \boldsymbol{\mu}_z \\ \mathbf{z}_i &= \boldsymbol{\mu}_z + \sqrt{L + k} \text{col}_i \mathbf{L} \\ \mathbf{z}_{i+L} &= \boldsymbol{\mu}_z - \sqrt{L + k} \text{col}_i \mathbf{L}, i = 1, \dots, L \end{aligned} \quad (1.26)$$

3. 对每个 sigmapoint 展开为状态和运动噪声的形式

$$\mathbf{z}_i = \begin{bmatrix} \hat{\mathbf{x}}_{k-1,i} \\ \mathbf{w}_{k,i} \end{bmatrix} \quad (1.27)$$

将每个 sigmapoint 代入非线性运动模型进行精确求解

$$\hat{\mathbf{x}}_{k,i} = \mathbf{f}(\hat{\mathbf{x}}_{k-1,i}, \mathbf{v}_k, \mathbf{w}_{k,i}), i = 0, \dots, 2L \quad (1.28)$$

4. 将转换后的 sigmapoint 重新组合成预测置信度：

$$\begin{aligned}
\check{\mathbf{x}}_k &= \sum_{i=0}^{2L} \alpha_i \check{\mathbf{x}}_{k,i} \\
\check{\mathbf{P}}_k &= \sum_{i=0}^{2L} \alpha_i (\check{\mathbf{x}}_{k,i} - \check{\mathbf{x}}_k)(\check{\mathbf{x}}_{k,i} - \check{\mathbf{x}}_k)^T \\
\alpha_i &= \begin{cases} \frac{k}{L+k}, i=0 \\ \frac{1}{2} \frac{1}{L+k}, \text{其他} \end{cases}
\end{aligned} \tag{1.29}$$

校正:

后验概率是 $\mathbf{p} \sim N(\hat{\mathbf{x}}_k, \hat{\mathbf{P}}_k)$

校正步骤主要是一次求解 $\mathbf{K}_k, \hat{\mathbf{P}}_k, \hat{\mathbf{x}}_k$ 。使用 SP 变换得到更优的 $\mu_{y,k}, \Sigma_{yy,k}, \Sigma_{xy,k}$ ，然后代入

到求解 $\mathbf{K}_k, \hat{\mathbf{P}}_k, \hat{\mathbf{x}}_k$ 的公式中去，进而得到后验值。采用下面的步骤:

1. 预测置信度和观测噪声堆叠在一起:

$$\mu_z = \begin{bmatrix} \check{\mathbf{x}}_k \\ 0 \end{bmatrix}, \Sigma_{zz} = \begin{bmatrix} \check{\mathbf{P}}_k & 0 \\ 0 & \mathbf{R}_k \end{bmatrix} \tag{1.30}$$

$$L = \dim(\mu_z) \tag{1.31}$$

2. 将 $\{\mu_z, \Sigma_{zz}\}$ 转化为 sigmapoint 表示:

$$\begin{aligned}
\mathbf{L}\mathbf{L}^T &= \Sigma_{zz} \\
\mathbf{z}_0 &= \mu_z \\
\mathbf{z}_i &= \mu_z + \sqrt{L+k} \mathbf{col}_i \mathbf{L} \\
\mathbf{z}_{i+L} &= \mu_z - \sqrt{L+k} \mathbf{col}_i \mathbf{L}, i=1, \dots, L
\end{aligned} \tag{1.32}$$

3. 对每个 sigmapoint 展开为状态和运动噪声的形式

$$\mathbf{z}_i = \begin{bmatrix} \check{\mathbf{x}}_{k,i} \\ \mathbf{n}_{k,i} \end{bmatrix} \tag{1.33}$$

将每个 sigmapoint 代入非线性运动模型进行精确求解

$$\check{\mathbf{y}}_{k,i} = \mathbf{f}(\check{\mathbf{x}}_{k,i}, \mathbf{n}_{k,i}), i=0, \dots, 2L \tag{1.34}$$

4. 将转换后的 sigmapoint 重新组合得到最终的结果:

$$\begin{aligned}
\mu_{y,k} &= \sum_{i=0}^{2L} \alpha_i^{\vee} y_{k,i} \\
\Sigma_{yy,k} &= \sum_{i=0}^{2L} \alpha_i^{\vee} (y_{k,i} - \mu_{y,k}^{\vee})(y_{k,i} - \mu_{y,k}^{\vee})^T \\
\Sigma_{xy,k} &= \sum_{i=0}^{2L} \alpha_i^{\vee} (x_{k,i}^{\vee} - x_k^{\vee})(y_{k,i}^{\vee} - \mu_{y,k}^{\vee})^T \\
\alpha_i &= \begin{cases} \frac{k}{L+k}, i=0 \\ \frac{1}{2} \frac{1}{L+k}, \text{其他} \end{cases}
\end{aligned} \tag{1.35}$$

SPKF 的优点：1) 不需要任何解析形式的导数 2) 仅使用了基本的线性代数运算，不需要非线性运动和观测模型的封闭形式（解析解）。

迭代 sigmapoint 卡尔曼滤波器（ISPKF）

ISPKF 比单次的版本更优秀。在每次迭代中，我们在工作点 $x_{op,k}$ 附近计算作为输入的

sigmapoint。第一次迭代时，令 $x_{op,k}^{\vee} = x_k^{\vee}$ ，但在之后每次迭代中 $x_{op,k}$ 都会被优化。

1. 预测置信度和观测噪声都有不确定性，将他们按以下方式堆叠：

$$\mu_z = \begin{bmatrix} x_{op,k} \\ 0 \end{bmatrix}, \Sigma_{zz} = \begin{bmatrix} P_k^{\vee} & 0 \\ 0 & R_k \end{bmatrix} \tag{1.36}$$

$$L = \dim(\mu_z) \tag{1.37}$$

2. 将 $\{\mu_z, \Sigma_{zz}\}$ 转化为 sigmapoint 表示：

$$\begin{aligned}
LL^T &= \Sigma_{zz} \\
z_0 &= \mu_z \\
z_i &= \mu_z + \sqrt{L+k} col_i L \\
z_{i+L} &= \mu_z - \sqrt{L+k} col_i L, i=1, \dots, L
\end{aligned} \tag{1.38}$$

3. 对每个 sigmapoint 展开为状态和观测噪声形式并代入非线性观测模型求解

$$z_i = \begin{bmatrix} x_{op,k,i} \\ n_{k,i} \end{bmatrix} \tag{1.39}$$

$$y_{op,k,i} = g(x_{op,k,i}, n_{k,i}), i=0, \dots, 2L \tag{1.40}$$

4.将转换后的 sigmapoint 重新组合得到最终的结果:

$$\begin{aligned}
\boldsymbol{\mu}_{y,k} &= \sum_{i=0}^{2L} \alpha_i \mathbf{y}_{op,k,i} \\
\Sigma_{yy,k} &= \sum_{i=0}^{2L} \alpha_i (\mathbf{y}_{op,k,i} - \boldsymbol{\mu}_{y,k})(\mathbf{y}_{op,k,i} - \boldsymbol{\mu}_{y,k})^T \\
\Sigma_{xy,k} &= \sum_{i=0}^{2L} \alpha_i (\mathbf{x}_{op,k,i} - \mathbf{x}_{op,k})(\mathbf{y}_{op,k,i} - \boldsymbol{\mu}_{y,k})^T \\
\Sigma_{xx,k} &= \sum_{i=0}^{2L} \alpha_i (\mathbf{x}_{op,k,i} - \mathbf{x}_{op,k})(\mathbf{x}_{op,k,i} - \mathbf{x}_{op,k})^T \\
\alpha_i &= \begin{cases} \frac{k}{L+k}, i=0 \\ \frac{1}{2} \frac{1}{L+k}, \text{其他} \end{cases}
\end{aligned} \tag{1.41}$$

然后求得

$$\begin{aligned}
\mathbf{K}_k &= \Sigma_{xy,k} \Sigma_{yy,k}^{-1} \\
\hat{\mathbf{P}}_k &= \Sigma_{xx,k} - \mathbf{K}_k \Sigma_{yx,k} \\
\hat{\mathbf{x}}_k &= \check{\mathbf{x}}_k + \mathbf{K}_k (\mathbf{y}_k - \boldsymbol{\mu}_{y,k} - \Sigma_{yx,k} \Sigma_{xx,k}^{-1} (\check{\mathbf{x}}_k - \mathbf{x}_{op,k}))
\end{aligned} \tag{1.42}$$

最初我们将工作点设置为先验的均值, $\mathbf{x}_{op,k} = \check{\mathbf{x}}_k$, 在随后的迭代中我们将其设置为当前迭

代的最优估计: $\mathbf{x}_{op,k} = \hat{\mathbf{x}}_k$, 当下一迭代量足够小时, 该过程停止。

IEKF 对应于 EKF。

ISPKF 对应于 SPKF, 令 $\mathbf{x}_{op,k} = \check{\mathbf{x}}_k$, 同样有

$$\hat{\mathbf{x}}_k = \check{\mathbf{x}}_k + \mathbf{K}_k (\mathbf{y}_k - \boldsymbol{\mu}_{y,k}) \tag{1.43}$$

粒子滤波器

粒子滤波器降低了卡尔曼滤波器的状态矩阵, 但是对于单个的计算单元增加了计算量。

滤波器的分类

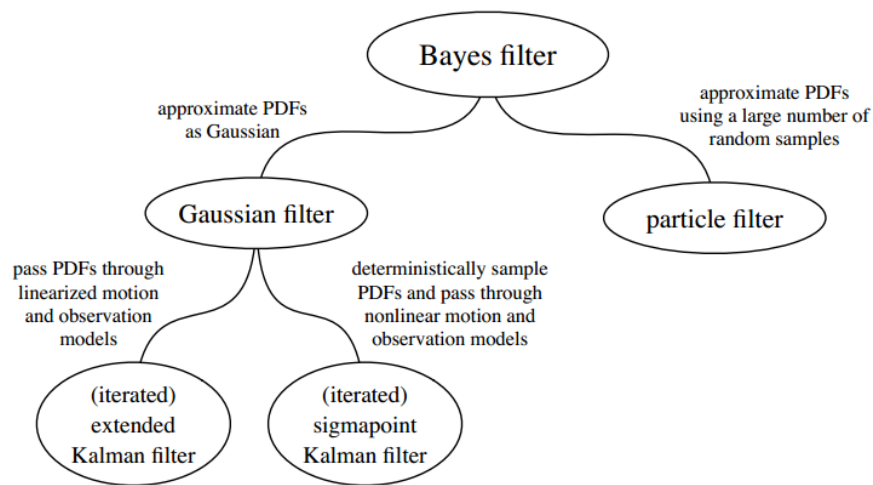


Figure 4.16
Taxonomy of the different filtering methods, showing their relationships to the Bayes filter.

上面的分类主要是根据将概率分布函数传入非线性函数的方法来分类的。线性化方法和 sigmapoint 方法是假设概率分布函数为高斯的来进行的。

参考文献

- [1] 概率机器人学
- [2] 机器人学的状态估计