

# 卡尔曼滤波的学习与应用

何富君编辑

## 目录

一 卡尔曼滤波的初步认识 .....	2
1.1 关于卡尔曼 .....	2
1.2 卡尔曼滤波器的举例说明 .....	2
1.3 卡尔曼滤波器算法 .....	3
1.4 简单例子 .....	3
二 卡尔曼滤波的实现 .....	5
2.1 卡尔曼滤波的基本假设: .....	5
2.2 卡尔曼滤波基本公式 .....	6
2.3 卡尔曼滤波参数的估计和调整 .....	7
2.4 卡尔曼滤波初始化 .....	8
3 扩展卡尔曼滤波 .....	10
3.1 扩展卡尔曼滤波的公式为: .....	11
3.2 另一种扩展卡尔曼滤波表示 .....	12
4 卡尔曼滤波的相关知识 .....	14
4.1 线性系统 .....	14
4.2 泰勒级数展开 .....	14
4.3 雅可比矩阵 .....	15
5 卡尔曼滤波应用 .....	16

# 一 卡尔曼滤波的初步认识

## 1.1 关于卡尔曼

在学习**卡尔曼滤波器**之前，首先看看为什么叫“卡尔曼”。跟其他著名的理论（例如傅立叶变换，泰勒级数等等）一样，卡尔曼也是一个人的名字，而跟他们不同的是，他是个现代人！

卡尔曼全名 Rudolf Emil Kalman，匈牙利数学家，1930 年出生于匈牙利首都布达佩斯。1953，1954 年于麻省理工学院分别获得电机工程学士及硕士学位。1957 年于哥伦比亚大学获得博士学位。我们现在要学习的**卡尔曼滤波器**，正是源于他的博士论文和 1960 年发表的论文《A New Approach to Linear Filtering and Prediction Problems》（线性滤波与预测问题的新方法）。如果对这篇论文有兴趣，可以到这里的地址下载：<http://www.cs.unc.edu/~welch/kalman/media/pdf/Kalman1960.pdf>

简单来说，**卡尔曼滤波器**是一个“optimal recursive data processing algorithm（最优化自回归数据处理算法）”。对于解决很大部分的问题，他是最优，效率最高甚至是最有用的。他的广泛应用已经超过 30 年，包括机器人导航，控制，传感器数据融合甚至在军事方面的雷达系统以及导弹追踪等等。近年来更被应用于计算机图像处理，例如头脸识别，图像分割，图像边缘检测等等。

## 1.2 卡尔曼滤波器的举例说明

为了可以更加容易的理解**卡尔曼滤波器**，这里会应用形象的描述方法来讲解，而不是像大多数参考书那样罗列一大堆的数学公式和数学符号。但是，他的 5 条公式是其核心内容。结合现代的计算机，其实卡尔曼的程序相当的简单，只要你理解了他的那 5 条公式。在介绍他的 5 条公式之前，先让我们来根据下面的例子一步一步的探索。

假设我们要研究的对象是一个房间的温度。根据你的经验判断，这个房间的温度是恒定的，也就是下一分钟的温度等于现在这一分钟的温度（假设我们用一分钟来做时间单位）。假设你对你的经验不是 100% 的相信，可能会有上下偏差几度。我们把这些偏差看成是高斯白噪声（White Gaussian Noise），也就是这些偏差跟前后时间是没有关系的而且符合高斯分配（Gaussian Distribution）。另外，我们在房间里放一个温度计，但是这个温度计也不准确的，测量值会比实际值偏差。我们也把这些偏差看成是高斯白噪声。

好了，现在对于某一分钟我们有两个有关于该房间的温度值：你根据经验的预测值（系统的预测值）和温度计的值（测量值）。下面我们要用这两个值结合他们各自的噪声来估算出房间的实际温度值。

假如我们要估算  $k$  时刻的是实际温度值。首先你要根据  $k-1$  时刻的温度值，来预测  $k$  时刻的温度。因为你相信温度是恒定的，所以你会得到  $k$  时刻的温度预测值是跟  $k-1$  时刻一样的，假设是 23 度，同时该值的高斯噪声的偏差是 5 度（5 是这样得到的：如果  $k-1$  时刻估算出的最优温度值的偏差是 3，你對自己预测的不确定度是 4 度，他们平方相加再开方，就是 5）。然后，你从温度计那里得到了  $k$  时刻的温度值，假设是 25 度，同时该值的偏差是 4 度。

由于我们用于估算  $k$  时刻的实际温度有两个温度值，分别是 23 度和 25 度。究竟实际温度是多少呢？相信自己还是相信温度计呢？究竟相信谁多一点，我们可以用他们的 covariance 来判断。因为  $K_g^2 = 5^2 / (5^2 + 4^2)$ ，所以  $K_g = 0.78$ ，我们可以估算出  $k$  时刻的实际温度值是： $23 + 0.78 * (25 - 23) = 24.56$  度。可以看出，因为温度计的 covariance 比较小（比较相信温度计），所以估算出的最优温度值偏向温度计的值。

现在我们已经得到  $k$  时刻的最优温度值了，下一步就是要进入  $k+1$  时刻，进行新的最优估算。到现在为止，好像还没看到什么自回归的东西出现。对了，在进入  $k+1$  时刻之前，我们还要算出  $k$  时刻那个最优值（24.56 度）的偏差。算法如下： $((1 - K_g) * 5^2)^{0.5} = 2.35$ 。这里的 5 就是上面的  $k$  时刻你预测的那个 23 度温度值的偏差，得出的 2.35 就是进入  $k+1$  时刻以后  $k$  时刻估算出的最优温度值的偏差（对应于上面的 3）。

就是这样，卡尔曼滤波器就不断的把 covariance 递归，从而估算出最优的温度值。他运行的很快，而且它只保留了上一时刻的 covariance。上面的  $K_g$ ，就是卡尔曼增益（Kalman Gain）。他可以随不同的时刻而改变他自己的值，是不是很神奇！

下面就要言归正传，讨论真正工程系统上的卡尔曼。

### 1.3 卡尔曼滤波器算法

(The Kalman Filter Algorithm)

在这一部分，我们就来描述源于 Dr Kalman 的卡尔曼滤波器。下面的描述，会涉及一些基本的概念知识，包括概率 (Probability)，随即变量 (Random Variable)，高斯或正态分配 (Gaussian Distribution) 还有 State-space Model 等等。但对于卡尔曼滤波器的详细证明，这里不能一一描述。

首先，我们先要引入一个离散控制过程的系统。该系统可用一个线性随机微分方程 (Linear Stochastic Difference equation) 来描述：

$$X(k)=A X(k-1)+B U(k)+W(k)$$

再加上系统的测量值：

$$Z(k)=H X(k)+V(k)$$

上两式子中， $X(k)$  是  $k$  时刻的系统状态， $U(k)$  是  $k$  时刻对系统的控制量。 $A$  和  $B$  是系统参数，对于多模型系统，他们为矩阵。 $Z(k)$  是  $k$  时刻的测量值， $H$  是测量系统的参数，对于多测量系统， $H$  为矩阵。 $W(k)$  和  $V(k)$  分别表示过程和测量的噪声。他们被假设成高斯白噪声 (White Gaussian Noise)，他们的 covariance 分别是  $Q$ ， $R$  (这里我们假设他们不随系统状态变化而变化)。

对于满足上面的条件 (线性随机微分系统，过程和测量都是高斯白噪声)，卡尔曼滤波器是最优的信息处理器。下面我们来用他们结合他们的 covariances 来估算系统的最优化输出 (类似上一节那个温度的例子)。

首先我们要利用系统的过程模型，来预测下一状态的系统。假设现在的系统状态是  $k$ ，根据系统的模型，可以基于系统的上一状态而预测出现在状态：

$$X(k|k-1)=A X(k-1|k-1)+B U(k) \dots\dots\dots (1)$$

式(1)中， $X(k|k-1)$  是利用上一状态预测的结果， $X(k-1|k-1)$  是上一状态最优的结果， $U(k)$  为现在状态的控制量，如果没有控制量，它可以为 0。

到现在为止，我们的系统结果已经更新了，可是，对应于  $X(k|k-1)$  的 covariance 还没更新。我们用  $P$  表示 covariance：

$$P(k|k-1)=A P(k-1|k-1) A' + Q \dots\dots\dots (2)$$

式(2)中， $P(k|k-1)$  是  $X(k|k-1)$  对应的 covariance， $P(k-1|k-1)$  是  $X(k-1|k-1)$  对应的 covariance， $A'$  表示  $A$  的转置矩阵， $Q$  是系统过程的 covariance。式子 1，2 就是卡尔曼滤波器 5 个公式当中的前两个，也就是对系统的预测。

现在我们有现在状态的预测结果，然后我们再收集现在状态的测量值。结合预测值和测量值，我们可以得到现在状态(k)的最优化估算值  $X(k|k)$ ：

$$X(k|k)= X(k|k-1)+K_g(k) (Z(k)-H X(k|k-1)) \dots\dots\dots (3)$$

其中  $K_g$  为卡尔曼增益 (Kalman Gain)：

$$K_g(k)= P(k|k-1) H' / (H P(k|k-1) H' + R) \dots\dots\dots (4)$$

到现在为止，我们已经得到了  $k$  状态下最优的估算值  $X(k|k)$ 。但是为了要另卡尔曼滤波器不断的运行下去直到系统过程结束，我们还要更新  $k$  状态下  $X(k|k)$  的 covariance：

$$P(k|k)= (I-K_g(k) H) P(k|k-1) \dots\dots\dots (5)$$

其中  $I$  为 1 的矩阵，对于单模型单测量， $I=1$ 。当系统进入  $k+1$  状态时， $P(k|k)$  就是式子(2)的  $P(k-1|k-1)$ 。这样，算法就可以自回归的运算下去。

卡尔曼滤波器的原理基本描述了，式子 1，2，3，4 和 5 就是他的 5 个基本公式。根据这 5 个公式，可以很容易的实现计算机的程序。下面，我会用程序举一个实际运行的例子。

### 1.4 简单例子

这里我们结合第二第三节，举一个非常简单的例子来说明卡尔曼滤波器的工作过程。所举的例子是进一步描述第二节的例子，而且还会配以程序模拟结果。

根据第二节的描述，把房间看成一个系统，然后对这个系统建模。当然，我们见的模型不需要非常地精确。我们所知道的这个房间的温度是跟前一时刻的温度相同的，所以  $A=1$ 。没有控制量，所以  $U(k)=0$ 。

因此得出：

$$X(k|k-1)=X(k-1|k-1) \dots\dots\dots (6)$$

式子（2）可以改成：

$$P(k|k-1)=P(k-1|k-1) + Q \dots\dots\dots (7)$$

因为测量的值是温度计的，跟温度直接对应，所以  $H=1$ 。式子 3, 4, 5 可以改成以下：

$$X(k|k)=X(k|k-1)+Kg(k) (Z(k)-X(k|k-1)) \dots\dots\dots (8)$$

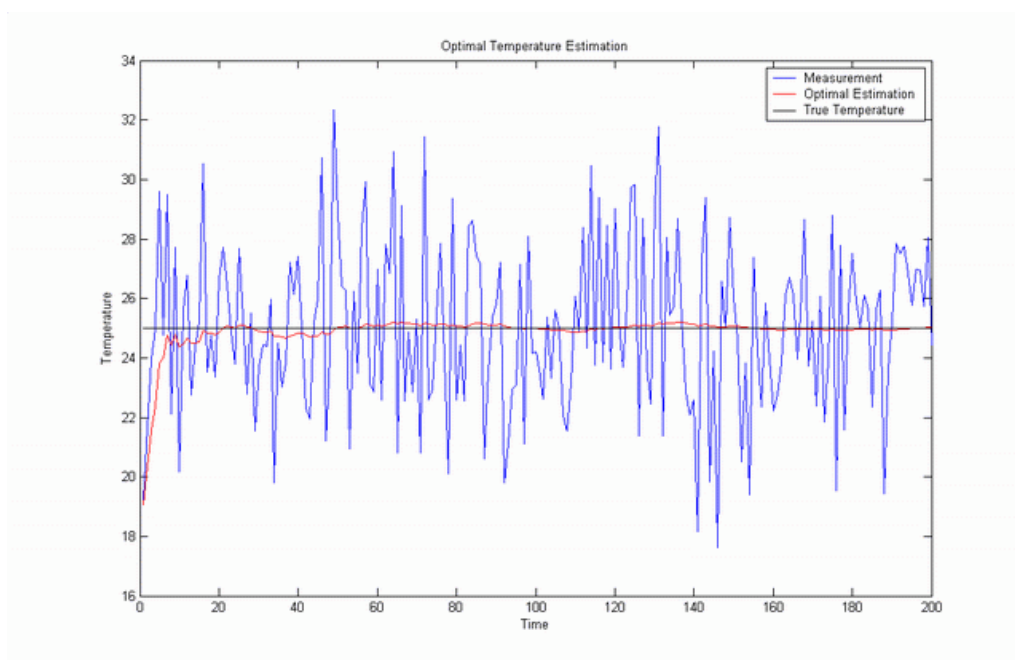
$$Kg(k)= P(k|k-1) / (P(k|k-1) + R) \dots\dots\dots (9)$$

$$P(k|k)= (1-Kg(k)) P(k|k-1) \dots\dots\dots (10)$$

现在我们模拟一组测量值作为输入。假设房间的真实温度为 25 度，我模拟了 200 个测量值，这些测量值的平均值为 25 度，但是加入了标准偏差为几度的高斯白噪声（在图中为蓝线）。

为了令卡尔曼滤波器开始工作，我们需要告诉卡尔曼两个零时刻的初始值，是  $X(0|0)$  和  $P(0|0)$ 。他们的值不用太在意，随便给一个就可以了，因为随着卡尔曼的工作， $X$  会逐渐的收敛。但是对于  $P$ ，一般不要取 0，因为这样可能会令卡尔曼完全相信你给定的  $X(0|0)$  是系统最优的，从而使算法不能收敛。我选了  $X(0|0)=1$  度， $P(0|0)=10$ 。

该系统的真实温度为 25 度，图中用黑线表示。图中红线是卡尔曼滤波器输出的最优化结果（该结果在算法中设置了  $Q=1e-6$ ,  $R=1e-1$ ）。



附 **matlab** 下面的 kalman 滤波程序：

```
clear
N=200;
w(1)=0;
w=randn(1,N)
x(1)=0;
```

```

a=1;
for k=2:N;
x(k)=a*x(k-1)+w(k-1);
end

V=randn(1,N);
q1=std(V);
Rvv=q1.^2;
q2=std(x);
Rxx=q2.^2;
q3=std(w);
Rww=q3.^2;
c=0.2;
Y=c*x+V;

p(1)=0;
s(1)=0;
for t=2:N;
p1(t)=a.^2*p(t-1)+Rww;
b(t)=c*p1(t)/(c.^2*p1(t)+Rvv);
s(t)=a*s(t-1)+b(t)*(Y(t)-a*c*s(t-1));
p(t)=p1(t)-c*b(t)*p1(t);
end

t=1:N;
plot(t,s,'r',t,Y,'g',t,x,'b');

```

以上内容来自：

杂八技术 - 卡尔曼滤波器（转载）

[http://blog.hexun.com/beverley\\_chou/1516094/rss/viewarticle.html](http://blog.hexun.com/beverley_chou/1516094/rss/viewarticle.html)

## 二 卡尔曼滤波的实现

这部分也属于数学结论的介绍，不进行理论推导，侧重于应用。

### 2.1 卡尔曼滤波的基本假设：

#### 1 适合于线性（随机）系统模型。

非线性的系统可以近似地线性化成为线性系统。所谓线性系统是符合叠加原理的系统，形式如下面的状态和测量方程。

#### 2 噪声为均值为 0 的高斯白噪声

白噪声：白噪声是指功率谱在整个频域内为常数的噪声，噪声是随机信号，因而白噪声没法求其频谱，只能求其功率谱。白噪声的自相关函数在  $t=0$  时不为 0，在  $t$  不等于 0 时值为零。“白色”是指噪声值与时间无关，不能通过现在时刻的噪声值预测其它时刻的噪声大小。白色也意味着噪声在所有的频率具有相等的功率谱。因为白噪声在不同时间的值是不相关的，所以任何两个不同时刻的噪声的协方差为 0，因此设状态方程的噪声  $W(k)$  为白噪声，则不同时刻的噪声向量的协方差  $E(W(k)W(j)^T) = Q(k)\delta_{kj}$ ，当  $k=j$  时，

$\delta_{ij} \neq 0$ ，当  $k \neq j$  时， $\delta_{ij} = 0$ 。即该协方差矩阵的主对角线元素不为零，其余的都为零。

高斯噪声是指噪声的分布为高斯分布，即正态分布。

## 2.2 卡尔曼滤波基本公式

设系统的状态方程为：

$$X(k) = AX(k-1) + BU(k-1) + W(k-1) \quad (1)$$

系统的测量方程：

$$Z(k) = CX(k) + V(k) \quad (2)$$

其中： $A(k-1)$ 为状态转移矩阵； $X(k-1)$ 为状态向量； $B(k-1)$ 为输入控制项矩阵； $u(k-1)$ 为输入或者控制信号； $W(k-1)$ 为零均值、白色高斯过程噪声序列，其协方差为  $Q(k-1)$ ； $C(k)$ 为测量矩阵； $V(k)$ 为协方差为  $R(k)$ 的零均值、白色高斯测量噪声。系统的测量方程的输出项  $Z(k)$ 是可以实际测量的量。

用  $\hat{X}(k)$  表示  $k$  时刻对随机信号  $X(k)$ 的最优线性滤波估计值，用  $\hat{X}(k+1|k)$  表示在  $k$  时刻对  $k+1$  时刻的信号  $X(k+1)$  的最优线性预测估计。

卡尔曼滤波器的递推算法如下：

状态的一步预测方程（基于系统的上一个状态）

$$\hat{X}(k|k-1) = A(k-1)\hat{X}(k-1|k-1) + B(k-1)U(k-1) \quad (a)$$

协方差的一步预测

$$P(k|k-1) = A(k-1)P(k-1|k-1)A^T(k-1) + Q(k-1) \quad (b)$$

滤波增益方程

$$K(k) = P(k|k-1)C^T(k)[C(k)P(k|k-1)C^T(k) + R(k)]^{-1} \quad (c)$$

滤波估计方程

$$\hat{X}(k|k) = \hat{X}(k|k-1) + K(k)[Z(k) - C(k)\hat{X}(k|k-1)] \quad (d)$$

滤波协方差更新方程

$$P(k|k) = [I - K(k)C(k)]P(k|k-1) \quad (e)$$

协方差的更新方程还有下面几种不同的形式：

$$P(k|k) = P(k|k-1) - K(k)C(k)P(k|k-1) \quad (e1)$$

式 (e) 和 (e1) 相同，适用于增益为最优卡尔曼增益的时候。使用其它增益的话要用公式 (e3)。

$$P(k|k) = P(k|k-1) - K(k)S(k)K^T(k) \quad (e2)$$

其中，测量的预测协方差（或新息协方差） $S(k) = C(k)P(k|k-1)C^T(k) + R(k)$ ，用来衡量新息（测量值减去测量估计值）的不确定性，新息的协方差越小，说明测量值越精确。（来源于《雷达数据处理及应用》，P24）。不知道公式 (e2) 有什么应用条件，但是应该比公式 (e1) 适应性好，该书中就使用该式进行卡尔曼滤波迭代。

$$P(k|k) = [I - K(k)C(k)]P(k|k-1)[I - K(k)C(k)]^T - K(k)R(k)K^T(k) \quad (e3)$$

其中， $\mathbf{I}$  为与协方差阵同维的单位阵。式 (e3) 可保证协方差阵  $\mathbf{P}$  的对称性和正定性。这个公式对于任何卡尔曼增益  $\mathbf{K}$  都成立。

式 (e) 或 (e1) 计算比较简单，所以实际中总是使用这个公式，但是需要注意这个公式仅在使用最优卡尔曼增益时它才成立。如果算术精度总是很低而导致 numerical stability 出现问题，或者特意使用了非最优卡尔曼增益，就必须使用式 (e3)。(引自《卡尔曼滤波 1—摘自博客》)，但是什么是最优卡尔曼增益还不大清楚，可以先试用简单的形式。

上面(a)~(e)的 5 个式子就是卡尔曼滤波的迭代公式。如果系统是非时变的，则系数矩阵  $\mathbf{A}$ 、 $\mathbf{B}$ 、 $\mathbf{C}$  都是常数矩阵。根据上面的 5 个公式就可以完成卡尔曼滤波的全过程并继续下去。因为把根据状态方程计算的预测值  $\hat{\mathbf{X}}(k|k-1)$  和  $k$  时刻的实际测量值  $\mathbf{Z}(k)$  相结合来估计  $k$  时刻的状态真值  $\mathbf{X}(k|k)$ ，所以起到了削减误差的最佳估计效果，实现了测量值的修正。

### 过程（状态）协方差矩阵 $\mathbf{P}$

两个标量  $x_1, x_2$  标准的协方差公式为  $\text{cov}(x_1, x_2) = E\{[x_1 - E(x_1)][x_2 - E(x_2)]\}$ 。

设状态向量  $\mathbf{X}$  有  $q$  个分量，一步滤波后各个状态分量的误差： $\mathbf{E}(k) = \mathbf{X}(k) - \hat{\mathbf{X}}(k)$ ，该误差的均方误差  $\mathbf{P}(k)$  就变成一个误差的协方差矩阵， $\mathbf{P}(k|k) = \text{cov}(\mathbf{X}(k) - \hat{\mathbf{X}}(k|k))$ ：

$$\mathbf{P}(k) = \begin{bmatrix} P_{11}(k) & P_{12}(k) & \cdots & P_{1q}(k) \\ P_{21}(k) & P_{22}(k) & \cdots & P_{2q}(k) \\ \vdots & \vdots & \cdots & \vdots \\ P_{q1}(k) & P_{q2}(k) & \cdots & P_{qq}(k) \end{bmatrix},$$

其中  $P_{ij}(k)$  为  $k$  时刻状态分量  $x_i$  和  $x_j$  的滤波误差的协方差：

$$P_{ij}(k) = E((x_i(k) - \hat{x}_i(k))(x_j(k) - \hat{x}_j(k))),$$

## 2.3 卡尔曼滤波参数的估计和调整

在卡尔曼滤波的使用中，通常首先要测量噪声协方差  $\mathbf{R}$ 。可以通过离线测量实验确定。但是确定过程噪声协方差  $\mathbf{Q}$  一般更困难，因为无法直接观察过程。当对一个相对简单的过程模型加入足够的不确定性时（通过选择合适的  $\mathbf{Q}$ ）得到的结果是可信的。无论如何，通过运行中调整  $\mathbf{Q}$  和  $\mathbf{R}$  可以得到较好的滤波效果。调整通常是离线的，通常借助于过程的另一个卡尔曼滤波进行系统辨识来实现。在  $\mathbf{Q}$  和  $\mathbf{R}$  为常数的情况下，估计误差协方差  $\mathbf{P}_k$  和卡尔曼增益  $\mathbf{K}_k$  都会迅速镇定，然后保持为常量。如果是这样的话，可以通过离线运行滤波器来预先确定这些参数，或者如 (Grewal and Andrews 2001) 所述来确定  $\mathbf{P}_k$  的稳态值。

然而通常的情况是测量误差不会保持常量。例如，当观察光电跟踪器面板上的亮点时，对附近的亮点的测量误差要小于远处的。有时过程噪声  $\mathbf{Q}$  在滤波器运行中是动态变化的，变成了  $\mathbf{Q}_k$ ，以调整符合不同的动态特点。例如在跟踪三维虚拟环境中的用户头部时，如果用户慢速移动，就要减小  $\mathbf{Q}_k$  值，如果动态变化加快就增加  $\mathbf{Q}_k$  值。在这类情况下，通过选择来适应用户意图的不确定性以及模型中的不确定性。(《An Introduction to the Kalman Filter》-35)

### 测量方程协方差的确定

测量方程是由测量设备可以实现的测量量来确定形式的，例如激光测距仪可以进行目标位置径向位置和角度位置的测量（极坐标测量），因此这两个量就是测量方程的输出。测量方程的协方差可以通过对激光测距仪的试验来确定。方法是利用在测距仪和目标都固定的情况下，激光测距仪对已知位置的目标进行多次测量，得到一组测量数据，然后利用 matlab 软件求出这组数据的协方差就可以作为测量方程的协方差

了。当然测距仪在测量不同距离的目标时，测量的误差可能不同，所以就有不同的协方差，可以取一个中间位置对应的协方差为仪器的协方差。或者取测量范围内最大的协方差。（来自纪老师）

### 状态方程（过程方程）的协方差确定

过程方程的协方差比较难以确定，一种方法是对于比较容易观察过程变化的系统，通过经验设定协方差。另一种需要通过仿真来调整。对于给定的过程方程，先预设一个协方差，通过设定一系列标准的测量值，仿真观察卡尔曼滤波是否很快收敛，如果不收敛或者收敛慢则可以调整过程协方差来达到较好的效果。（纪军红）

\*\*\*\*\*过程、测量误差及协方差的确定

误差及协方差的准确确定比较重要，也比较困难。测量方程的误差可以通过离线测量确定，但是过程误差不容易准确估计。

设状态方程  $X(k+1) = A(k)X(k) + B(k)u(k) + V(k)$ ， $V(k)$  为零均值、白色高斯过程噪声，其协方差为  $Q(k)$ 。如果噪声  $V(k)$  表示成  $\Gamma(k)v(k)$ ， $\Gamma(k)$  为过程噪声分布矩阵， $v(k)$  为一个标量。则  $Q(k)$  变为  $\Gamma(k)\sigma_v^2(k)\Gamma^T(k)$ 。以目标在  $x$  轴方向上做匀速直线运动为例：状态向量  $X(k) = [x, \dot{x}]^T$ ，噪声分布矩

阵为  $\Gamma(k) = \begin{bmatrix} T^2/2 \\ T \end{bmatrix}$ ， $T$  为采样周期，而标量  $v(k)$  是 0 均值，方差为  $q$  的量。 $E[v^2(k)] = q$ 。为何噪声分

布矩阵取这样的值？因为这是过程噪声，两个过程变量分别为目标移动的位置和速度，它们的过程噪声有一定的相关性。因为设定目标为匀速运动，所以过程中产生加速度的变化引起状态变量的噪声。

$\Delta x = aT^2/2$ ， $\Delta \dot{x} = aT$ ，这里将认为  $v(k) = a$ ，是方差为  $q$  的零均值变量， $q$  的值可以参照在加速度测量中的加速度变化情况。当然还可能存在其它噪声形式，需要通过工作过程中的系统辨识等方法来确定。

协方差的计算。 $Q(k)$  是噪声  $V(k)$  的协方差。

$$Q(k) = E[V(k)V^T(k)] = \begin{bmatrix} E[v_1^2(k)] & E[v_1(k)v_2(k)] \\ E[v_2(k)v_1(k)] & E[v_2^2(k)] \end{bmatrix}$$

如果噪声向量中各分量都服从各自的 0 均值正态分布： $v_1(k) \sim N(0, r_1^2)$ ， $v_2(k) \sim N(0, r_2^2)$ 。

其中， $E[v_1^2(k)] = E[(v_1(k) - 0)^2] = E[(v_1(k) - E(v_1(k)))^2] = r_1^2$

$$E[v_1(k)v_2(k)] = E[(v_1(k) - E(v_1(k)))(v_2(k) - E(v_2(k)))] =$$

如果噪声向量的各分量互不相关，则  $E[v_1(k)v_2(k)] = 0$ 。

## 2.4 卡尔曼滤波初始化

初始化也比较重要，初始化偏差大容易振动，甚至可能不收敛（？）。

例如跟踪一辆车（见《卡尔曼滤波 1—摘自博客》），状态空间为  $X_k = [x \quad \dot{x}]^T$ ，如果知道足够精确的车的初始位置，那么可以初始化  $\hat{X}(0|0) = [0 \quad 0]$  或者别的具体值。并且，通过给状态协方差矩阵赋较小



的值来告诉滤波器我们知道确切的初始位置，例如初始化为  $P(0|0) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$ ，或者给定其它较小的值。

如果我们不确切的知道最初的位置与速度，那么协方差矩阵可以初始化为一个对角线元素是  $b$  的矩阵， $b$  取一个合适的比较大的数。  $P(0|0) = \begin{bmatrix} b & 0 \\ 0 & b \end{bmatrix}$ 。此时，滤波器更倾向于使用初次测量值的信息。

## 1 二维状态向量估计-从《雷达数据处理及应用》P34 页中

设雷达进行一维跟踪时，状态方程为  $X_k = [x \quad \dot{x}]^T$ ，测量噪声  $V(k) \sim N(0, r)$ （即  $V$  符合均值为 0，方差为  $r$  的正态分布），且与过程噪声独立。这时状态估计初始化可以采用初始两点差分，利用第一和第二时刻的两个测量值  $Z(0)$  和  $Z(1)$  进行初始化。

$$\hat{X}(1|1) = \begin{bmatrix} \hat{x}(1|1) \\ \hat{\dot{x}}(1|1) \end{bmatrix} = \begin{bmatrix} z(1) \\ \frac{z(1) - z(0)}{T} \end{bmatrix},$$

初始协方差为：  $P(1|1) = \begin{bmatrix} r & r/T \\ r/T & 2r/T^2 \end{bmatrix}$

$T$  为测量的采样周期。状态估计和滤波从  $k=2$  时刻开始。

## 2 四维状态向量估计的初始化

属于两坐标雷达的数据处理，系统的状态向量表示为  $X(k) = [x \quad \dot{x} \quad y \quad \dot{y}]^T$ ，

状态方程为：  $X(k) = AX(k-1) + W(k-1)$ ，测量方程为：  $Z(k) = CX(k) + V(k)$ ，测量状态向量：

$$Z(k) = \begin{bmatrix} z_1(k) \\ z_2(k) \end{bmatrix} = \begin{bmatrix} x(k) \\ y(k) \end{bmatrix} = \begin{bmatrix} \rho \cos \theta \\ \rho \sin \theta \end{bmatrix}。 \quad (a)$$

注意这里选择直角坐标系表示目标状态数据，但是雷达的测量数据是以极坐标形式表示的，这里在测量变量里把极坐标数据转化为直角坐标数据，这样做的目的或许是把测量方程也变成了线性方程，这样的话，就可以用普通的卡尔曼滤波方法。否则的话，测量向量以极坐标表示，写成

$$Z(k) = \begin{bmatrix} z_1(k) \\ z_2(k) \end{bmatrix} = \begin{bmatrix} \rho \\ \theta \end{bmatrix} = \begin{bmatrix} \sqrt{x^2 + y^2} \\ a \tan(y/x) \end{bmatrix}, \quad (b)$$

是非线性方程，只能写成  $Z(k) = f(X(k)) + V(k)$  的形式，需要用扩展卡尔曼滤波方法。

系统的初始状态可利用前两个时刻的测量值  $z(0)$  和  $z(1)$  来确定，即

$$X(1|1) = \begin{bmatrix} z_1(1) & \frac{z_1(1) - z_1(0)}{T} & z_2(1) & \frac{z_2(1) - z_2(0)}{T} \end{bmatrix}^T \quad (c)$$

由于初始状态值是由测量值给定的，所以初始状态协方差矩阵  $P$  也是由测量方差相应地设定。由于测量方程中的测量量（即输出量）是由真实测量值  $\rho$  和  $\theta$  转化来的，所以测量  $Z$  的测量噪声也需要由直接测量值  $\rho$  和  $\theta$  的测量噪声求出。

$k$  时刻测量噪声在直角坐标系下的协方差为：

$$R(k) = \begin{bmatrix} r_{11} & r_{12} \\ r_{12} & r_{22} \end{bmatrix} = A \begin{bmatrix} \sigma_\rho^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix} A^T \quad (d)$$

其中： $\sigma_\rho^2$  和  $\sigma_\theta^2$  分别为径向距离和方位角测量误差的协方差。即  $\rho$  的测量误差服从  $N(0, \sigma_\rho^2)$  分布。而  $A$  为  $Z$  对  $\rho$  和  $\theta$  的雅可比矩阵，但是式 (d) 是如何推出来的？

$$A = \begin{bmatrix} \cos \theta & -\rho \sin \theta \\ \sin \theta & \rho \cos \theta \end{bmatrix}, \quad (e)$$

则初始的状态协方差阵可写成：

$$P(1|1) = \begin{bmatrix} r_{11}(1) & r_{11}(1)/T & r_{12}(1) & r_{12}(1)/T \\ r_{11}(1)/T & 2r_{11}(1)/T^2 & r_{12}(1)/T & 2r_{12}(1)/T^2 \\ r_{12}(1) & r_{12}(1)/T & r_{22}(1) & r_{22}(1)/T \\ r_{12}(1)/T & 2r_{12}(1)/T^2 & r_{22}(1)/T & 2r_{22}(1)/T^2 \end{bmatrix} \quad (f)$$

滤波器从  $k=2$  时刻开始工作。

当然如果测量方程的输出直接就是  $\rho$  和  $\theta$ ，测量向量是式 (b) 的形式，而状态方程不变，那么初始的状态协方差阵可能仍然要写成 (f) 的形式，因为状态量仍然是从测量量转化过来的，它们之间的关系没有变。

但是如果状态方程发生改变，直接以  $\rho$  和  $\theta$  以及它们的速度为状态向量，测量方程也是以  $\rho$  和  $\theta$  输出：

$$X(k) = [x \quad \dot{x} \quad y \quad \dot{y}]^T, \quad Z(k) = [\rho \quad \theta]^T,$$

状态方程为： $X(k) = AX(k-1) + W(k-1)$ ，测量方程为： $Z(k) = CX(k) + V(k)$

测量噪声  $V(k)$  为均值为零，协方差为  $R(k)$ 。

则  $R(k) = \begin{bmatrix} r_{11} & r_{12} \\ r_{12} & r_{22} \end{bmatrix} = \begin{bmatrix} \sigma_\rho^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix}$ ， $\sigma_\rho^2$  和  $\sigma_\theta^2$  分别为径向距离和方位角测量误差的协方差。

则初始的状态协方差阵可写成：

$$P(1|1) = \begin{bmatrix} r_{11}(1) & r_{11}(1)/T & r_{12}(1) & r_{12}(1)/T \\ r_{11}(1)/T & 2r_{11}(1)/T^2 & r_{12}(1)/T & 2r_{12}(1)/T^2 \\ r_{12}(1) & r_{12}(1)/T & r_{22}(1) & r_{22}(1)/T \\ r_{12}(1)/T & 2r_{12}(1)/T^2 & r_{22}(1)/T & 2r_{22}(1)/T^2 \end{bmatrix} = \begin{bmatrix} \sigma_\rho^2 & \sigma_\rho^2/T & 0 & 0 \\ \sigma_\rho^2/T & 2\sigma_\rho^2/T^2 & 0 & 0 \\ 0 & 0 & \sigma_\theta^2 & \sigma_\theta^2/T \\ 0 & 0 & \sigma_\theta^2/T & 2\sigma_\theta^2/T^2 \end{bmatrix}$$

### 3 扩展卡尔曼滤波

卡尔曼滤波是在线性高斯情况下利用最小均方误差准则获得目标的动态估计，适应于过程和测量都属于线性系统，且误差符合高斯分布的系统。但是实际上很多系统都存在一定的非线性，表现在过程方程（状态方程）是非线性的，或者观测与状态之间的关系（测量方程）是非线性的。这种情况下就不能使用一般

的卡尔曼滤波了。解决的方法是将非线性关系进行线性近似，进而转化成线性问题。最常用的线性化方法是泰勒级数展开，忽略高次项。在此基础上的卡尔曼滤波就是扩展卡尔曼滤波（EKF）。泰勒展开一般可以取一次项或者二次项（忽略三次以后的高次项）。仿真结果表明，二阶扩展卡尔曼滤波的性能远比一阶的好，而二阶以上的扩展卡尔曼滤波性能与二阶相比没有明显提高，所以一般不采用。但是二阶的计算量很大，所以一般采用一阶卡尔曼滤波。（《雷达数据处理及应用》P45）

由于状态方程或测量方程可能是非线性的，所以不能写成标准的线性化的格式，所以记成如下的表达式形式，当然对于实际的系统就需要写出真实的状态方程和测量方程表达式，而不需要整理成标准的线性化格式。（见《An introduction to the Kalman Filter》-25 页，为了与前面的书写方式相同，对这里公式的书写和符号的标记方式进行了修改）

下面给出经过一阶泰勒展开线性化的卡尔曼滤波递推公式。（见《An introduction to the Kalman Filter》-25）  
状态方程：（直接写成函数的形式）

$$\mathbf{x}(k) = f(\mathbf{x}(k-1), \mathbf{u}(k-1), \mathbf{w}(k-1))$$

测量方程：

$$\mathbf{z}(k) = h(\mathbf{x}(k), \mathbf{v}(k))$$

### 3.1 扩展卡尔曼滤波的公式为：

状态的一步预测方程（基于系统的上一个状态）

$$\hat{\mathbf{x}}(k|k-1) = f(\mathbf{x}(k-1|k-1), \mathbf{u}(k-1), 0) \quad (1)$$

协方差的一步预测

$$\mathbf{P}(k|k-1) = \mathbf{A}(k-1)\mathbf{P}(k-1|k-1)\mathbf{A}^T(k-1) + \mathbf{F}(k-1)\mathbf{Q}(k-1)\mathbf{F}^T(k-1) \quad (2)$$

滤波增益方程

$$\mathbf{K}(k) = \mathbf{P}(k|k-1)\mathbf{H}^T(k)[\mathbf{H}(k)\mathbf{P}(k|k-1)\mathbf{H}^T(k) + \mathbf{G}(k)\mathbf{R}(k)\mathbf{G}^T(k)]^{-1} \quad (3)$$

滤波估计方程

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{K}(k)[\mathbf{z}(k) - h(\hat{\mathbf{x}}(k|k-1), 0)] \quad (4)$$

滤波协方差更新方程

$$\mathbf{P}(k|k) = (\mathbf{I} - \mathbf{K}(k)\mathbf{H}(k))\mathbf{P}(k|k-1) \quad (5)$$

式中：

$\mathbf{A}$  是函数  $f$  对  $\mathbf{x}$  的偏微分雅可比矩阵。  $\mathbf{A}_{[i,j]} = \frac{\partial f_{[i]}}{\partial \mathbf{x}_{[j]}}(\hat{\mathbf{x}}(k-1|k-1), \mathbf{u}(k-1), 0)$ ,

$\mathbf{F}$  是  $f$  对噪声  $\mathbf{w}$  的偏微分雅可比矩阵。  $\mathbf{F}_{[i,j]} = \frac{\partial f_{[i]}}{\partial \mathbf{w}_{[j]}}(\hat{\mathbf{x}}(k-1|k-1), \mathbf{u}(k-1), 0)$ ,

$\mathbf{H}$  是  $h$  对  $\mathbf{x}$  的偏微分雅可比矩阵。  $\mathbf{H}_{[i,j]} = \frac{\partial h_{[i]}}{\partial \mathbf{x}_{[j]}}(\hat{\mathbf{x}}(k|k-1), 0)$ ,

$\mathbf{G}$  是  $h$  对  $\mathbf{v}$  的偏微分雅可比矩阵。  $\mathbf{G}_{[i,j]} = \frac{\partial h_{[i]}}{\partial \mathbf{v}_{[j]}}(\hat{\mathbf{x}}(k|k-1), 0)$ ,

$Q(k-1)$  是噪声  $w(k-1)$  的协方差,

$R(k)$  是噪声  $v(k)$  的协方差。

### 3.2 另一种扩展卡尔曼滤波表示

(引自《雷达数据处理及应用》)

无论状态方程或者测量方程是非线性的, 都需要使用扩展卡尔曼滤波方法。这种方法就是对非线性方程进行线性化。

非线性系统的状态方程为 (忽略系统输入):

$$X(k+1) = f(k, X(k)) + V(k)$$

测量方程为:

$$z(k) = h(k, X(k)) + W(k)$$

一阶泰勒展开线性化的扩展卡尔曼滤波的迭代公式为:

状态的一步预测

$$\hat{X}(k+1|k) = f(k, \hat{X}(k|k)) \quad (1)$$

协方差的一步预测

$$P(k+1|k) = f_x(k)P(k|k)f_x^T(k) + Q(k) \quad (2)$$

测量预测值

$$\hat{z}(k+1|k) = h(k+1, \hat{X}(k+1|k)) \quad (3)$$

它相伴的协方差 (新息协方差或测量的预测协方差)

$$S(k+1) = h_x(k+1)P(k+1|k)h_x^T(k+1) + R(k+1) \quad (4)$$

增益更新方程

$$K(k+1) = P(k+1|k)h_x^T(k+1)S^{-1}(k+1) \quad (5)$$

状态更新方程

$$\hat{X}(k+1|k+1) = \hat{X}(k+1|k) + K(k+1)[z(k+1) - \hat{z}(k+1|k)] \quad (6)$$

协方差更新方程

$$P(k+1|k+1) = [I - K(k+1)h_x(k+1)]P(k+1|k)[I + K(k+1)h_x(k+1)]^T - K(k+1)R(k+1)K^T(k+1) \quad (7)$$

$$P(k+1|k+1) = [I - K(k+1)h_x(k+1)]P(k+1|k) \quad (8)$$

对于更新协方差, 使用式 (7) 是对任何增益  $k$  都适用, 但是如果  $k(k+1)$  为最优卡尔曼增益则可以使用式 (8)。在一般情况下可以优先使用式 (8)。

高斯零均值白噪声的方差为:  $E[V(k)V^T(j)] = Q(k)\delta_{kj}$

高斯零均值白噪声的方差为：  $E[W(k)W^T(j)] = R(k)\delta_{kj}$

利用泰勒级数展开，首先对状态方程在定值  $\hat{X}(k|k)$ （ $k$  时刻的状态估计值）附近进行泰勒一阶级数展开。

$$X(k+1) = f(k, \hat{X}(k|k)) + f_x(k)[X(k) - \hat{X}(k|k)] + V(k)$$

其中  $f_x(k)$  是  $f$  对状态量  $X$  的雅可比矩阵，

$$f_x(k) = [\nabla_x f'(k, X)]'_{X=\hat{X}(k|k)} = \begin{bmatrix} \frac{\partial}{\partial x_1} \\ \vdots \\ \frac{\partial}{\partial x_n} \end{bmatrix} [f_1(X) \quad \cdots \quad f_n(X)]_{X=\hat{X}(k|k)} = \begin{bmatrix} \frac{\partial f_1(X)}{\partial x_1} & \cdots & \frac{\partial f_1(X)}{\partial x_n} \\ \vdots & \cdots & \vdots \\ \frac{\partial f_n(X)}{\partial x_1} & \cdots & \frac{\partial f_n(X)}{\partial x_n} \end{bmatrix}_{X=\hat{X}(k|k)}$$

式中  $x_1, x_2 \cdots x_n$  为状态向量  $X(k)$  的元素，而  $f_1(X), f_2(X) \cdots f_n(X)$  则为  $n$  个状态方程（不是写成向量表示的状态方程）。

两种扩展卡尔曼滤波的表示方法基本一致，第二种表示方法看似多了几个公式，实际上是将第一种表示方法中的一个表达式拆成了两个，看起来清晰些。

同理，  $h_x(k+1)$  是  $h$  对状态量  $X$  的雅可比矩阵，

$$h_x(k+1) = [\nabla_x h'(k+1, X)]'_{X=\hat{X}(k+1|k)} = \begin{bmatrix} \frac{\partial h_1(X)}{\partial x_1} & \cdots & \frac{\partial h_1(X)}{\partial x_n} \\ \vdots & \cdots & \vdots \\ \frac{\partial h_n(X)}{\partial x_1} & \cdots & \frac{\partial h_n(X)}{\partial x_n} \end{bmatrix}_{X=\hat{X}(k+1|k)}$$

### 3.3 线性化 EKF 滤波的误差补偿及注意事项

因为扩展卡尔曼滤波算法是由泰勒级数的一阶或二阶展开式获得，并忽略了高阶项，这样在滤波过程中要引入一定的线性化误差，可采用以下补偿方法[14,17]:

1 为补偿状态预测中的误差，附加“人为过程噪声”，即通过增大过程噪声协方差来实现这一点。

2 用标量加边因子  $\phi > 1$  乘状态预测协方差矩阵，即  $P^*(k+1|k) = \phi P(k+1|k)$ ，然后在协方差更新方程中使用  $P^*(k+1|k)$ 。

3 利用对角矩阵  $\phi = \text{diag}(\sqrt{\phi_i}), \phi_i > 1$  乘状态预测协方差矩阵，即

$$P^*(k+1|k) = \phi' P(k+1|k) \phi$$

4 采用迭代滤波，即通过平滑技术改进参考估计来降低线性化误差。

扩展卡尔曼滤波应用中应该注意的一些问题

扩展卡尔曼滤波是比较常用的非线性滤波方法，在这种方法中，非线性因子的存在对滤波稳定性和状

态估计精度都有很大的影响，其滤波结果的好坏与过程噪声和测量噪声的统计特性又有很大关系。由于扩展卡尔曼滤波中预先估计的过程噪声协方差  $Q(k)$  和测量噪声协方差  $R(k)$  在滤波过程中一直保持不变，如果这两个噪声协方差矩阵估计的不太准确的话，这样在滤波过程中就容易产生误差积累，导致滤波发散，而且对于维数较大的非线性系统，估计的过程噪声协方差矩阵和测量噪声协方差矩阵易出现异常现象，即  $Q(k)$  失去半正定性， $R(k)$  失去正定性，也容易导致滤波发散。利用 EKF 对目标进行跟踪，只有当系统的动态模型和观测模型都接近线性时，也就是线性化模型误差较小时，扩展卡尔曼滤波的结果才有可能接近于真实值。而且扩展卡尔曼滤波还有一个缺点就是状态的初始值不太好确定，如果假设的状态初始值和初始协方差误差较大的话，也容易导致滤波发散。

## 4 卡尔曼滤波的相关知识

### 4.1 线性系统

对于线性系统的判断。当描述动态系统的数学方程具有线性属性时，称相应的系统为线性系统。线性系统的基本特征是满足叠加原理，即若表示系统的数学描述为  $L$ ，那么对任意两个输入变量  $u_1$  和  $u_2$  以及任意两个有限常数  $c_1$  和  $c_2$ ，必有  $L(c_1u_1 + c_2u_2) = c_1L(u_1) + c_2L(u_2)$ 。（《线性系统理论-段广仁-5》）。

从网上得到如下说明：状态变量和输出变量对于所有可能的输入变量和初始状态都满足叠加原理的系统。叠加原理是指：如果系统相应于任意两种输入和初始状态  $(u_1, x_{01})$  和  $(u_2, x_{02})$  时的状态和输出分别为  $(x_1(t), y_1(t))$  和  $(x_2(t), y_2(t))$ ，则当输入和初始状态为  $(C_1u_1(t) + C_2u_2(t), C_1x_{01} + C_2x_{02})$  时，系统的状态和输出必为  $(C_1x_1(t) + C_2x_2(t), C_1y_1(t) + C_2y_2(t))$ ，其中  $x$  表示状态， $y$  表示输入， $u$  表示输出， $C_1$  和  $C_2$  为任意实数。一个由线性元部件所组成的系统必是线性系统。但是，相反的命题在某些情况下可能不成立。线性系统的状态变量(或输出变量)与输入变量间的因果关系可用一组线性微分方程或差分方程来描述，这种方程称为系统的数学模型。作为叠加性质的直接结果，线性系统的一个重要性质是系统的响应可以分解为两个部分：零输入响应和零状态响应。前者指由非零初始状态所引起的响应；后者则指由输入引起的响应。两者可分别计算。这一性质为线性系统的分析和研究带来很大方便。

### 4.2 泰勒级数展开

设函数  $f(x)$  在  $x=a$  处展开，展开公式为：

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots + \frac{f^n(a)}{n!}(x-a)^n$$

对于非线性系统的状态方程为：

$$X(k+1) = f(k, X(k)) + W(k), \text{ 其中输暂无输入项。}$$

为了得到一步预测状态  $X(k+1|k)$ ，对上式中的非线性函数在  $X(k|k)$ （已经是一个具体的数值）附近进行泰勒级数展开，取其一阶项。完整的泰勒级数展开为：（见《雷达数据处理及应用》-43）

$$\begin{aligned} X(k+1) = & f(k, X(k|k)) + f_x(k)[X(k) - X(k|k)] \\ & + \frac{1}{2} \sum_{i=1}^n e_i [X(k) - X(k|k)]^T f_{xx}^i(k) [X(k) - X(k|k)] + (\text{高阶项}) + W(k) \end{aligned}$$

其中： $n$  为状态向量的维数， $e_i$  为第  $i$  个笛卡尔基本向量，例如四维向量情况下的笛卡尔基本向量有 4 个：

$$e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad e_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad e_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \quad e_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

并且  $f(k, X(k))$  的一阶导数  $f_X(k)$  :

$$f_X(k) = [\nabla_X f'(k, X)]'_{X=X(k|k)} = \begin{bmatrix} \frac{\partial}{\partial x_1} \\ \vdots \\ \frac{\partial}{\partial x_n} \end{bmatrix} [f_1(X) \quad \cdots \quad f_n(X)]_{X=X(k|k)}$$

$$= \begin{bmatrix} \frac{\partial f_1(X)}{\partial x_1} & \cdots & \frac{\partial f_n(X)}{\partial x_1} \\ \vdots & \cdots & \vdots \\ \frac{\partial f_1(X)}{\partial x_n} & \cdots & \frac{\partial f_n(X)}{\partial x_n} \end{bmatrix}_{X=X(k|k)}$$

也是向量  $f$  的雅可比矩阵，在状态的最近估计  $X(k|k)$  上取值。

类似地可求向量  $f$  的第  $i$  个分量的海赛矩阵为：

$$f_{XX}^i(k) = [\nabla_X \nabla_X' f^i(k, X)]_{X=X(k|k)} = \begin{bmatrix} \frac{\partial^2 f^i(X)}{\partial x_1 \partial x_1} & \cdots & \frac{\partial^2 f^i(X)}{\partial x_1 \partial x_n} \\ \vdots & \cdots & \vdots \\ \frac{\partial^2 f^i(X)}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f^i(X)}{\partial x_n \partial x_n} \end{bmatrix}_{X=X(k|k)}$$

可见二阶泰勒展开的计算量很大，所以一般取其一阶展开。状态方程的一阶泰勒展开线性化方程为：

$$X(k+1) = f(k, X(k|k)) + f_X(k)[X(k) - X(k|k)] + W(k)$$

对一个函数进行泰勒级数一阶展开就是对该函数进行了线性化，如果取其二阶展开则只是对原来非线性函数的另一种形式的近似，仍然是非线性的。

### 4.3 雅可比矩阵

雅可比（Jacobi）矩阵在各种场合应用很多。雅可比就是输出量与输入量的微分比，表明了输入量的变化（微分）对输出量变化的影响。

对于单变量函数  $y = f(x)$ ，雅可比  $J(x) = dy/dx$ ，即  $dy = J(x)dx$ ，所以雅可比实际上也就是输出量对输入量的一阶导数。

对于输入量和输出量都是向量的函数关系，雅可比就变成了矩阵，称为雅可比矩阵，同样表示的是输出量与输入量之间的微分关系。设  $\mathbf{y} = f(\mathbf{x})$ ，其中  $\mathbf{x} = (x_1, \cdots, x_n)^T$ ， $\mathbf{y} = (y_1, \cdots, y_m)^T$ 。则  $\mathbf{y}$  对  $\mathbf{x}$  的雅可比矩阵为：

$$J(\mathbf{x}) = \frac{d\mathbf{y}}{d\mathbf{x}^T} \begin{bmatrix} \frac{\partial \mathbf{y}}{\partial x_1} & \frac{\partial \mathbf{y}}{\partial x_2} & \cdots & \frac{\partial \mathbf{y}}{\partial x_n} \end{bmatrix} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} & \cdots & \frac{\partial y_2}{\partial x_n} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial y_m}{\partial x_1} & \frac{\partial y_m}{\partial x_2} & \cdots & \frac{\partial y_m}{\partial x_n} \end{bmatrix} = \left( \frac{\partial y_i}{\partial x_j} \right)_{m \times n}$$

在机器人系统中，操作臂的运动方程  $\mathbf{x} = f(\mathbf{q})$  代表操作空间  $\mathbf{x}$  与关节空间  $\mathbf{q}$  之间的位移关系。将式子两边对时间求导，即得出  $\mathbf{q}$  与  $\mathbf{x}$  的微分关系。 $\dot{\mathbf{x}} = J(\mathbf{q})\dot{\mathbf{q}}$ ，代表操作空间与关节空间的速度关系。 $J(\mathbf{q})$  称为速度雅可比矩阵。

可以看出雅可比矩阵实际上就是输出量对输入量的一阶导数，所以可以直接针对一个函数关系写出相应的雅可比矩阵。

对于直角坐标与极坐标的变换关系  $\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \rho \cos \theta \\ \rho \sin \theta \end{bmatrix}$ ，可以简写成  $\mathbf{z} = f(\mathbf{q})$ ，其中，

$\mathbf{q} = [\rho \quad \theta]^T$ ，那么直接写出  $\mathbf{z}$  对  $\mathbf{q}$  的雅可比矩阵：

$$J(\mathbf{q}) = \frac{\partial f_i(\mathbf{q})}{\partial q_j} = \begin{bmatrix} \frac{\partial f_1(\mathbf{q})}{\partial q_1} & \frac{\partial f_1(\mathbf{q})}{\partial q_2} \\ \frac{\partial f_2(\mathbf{q})}{\partial q_1} & \frac{\partial f_2(\mathbf{q})}{\partial q_2} \end{bmatrix} = \begin{bmatrix} \frac{\partial z_1}{\partial \rho} & \frac{\partial z_1}{\partial \theta} \\ \frac{\partial z_2}{\partial \rho} & \frac{\partial z_2}{\partial \theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & -\rho \sin \theta \\ \sin \theta & \rho \cos \theta \end{bmatrix}$$

$\mathbf{z}$  和  $\mathbf{q}$  之间是非线性关系，如果该方程应用于卡尔曼滤波，需要对方程进行线性化，利用泰勒一阶展开进行线性化。假设对值  $\mathbf{q}_k = [\rho_k \quad \theta_k]^T$  线性化。

$$X(k+1) = f(k, X(k|k)) + f_x(k)[X(k) - X(k|k)]$$

$$\mathbf{z}_{k+1} = f(\mathbf{q}_k) + J(\mathbf{q}_k)(\mathbf{q}_{k+1} - \mathbf{q}_k)$$

$$\mathbf{z}_{k+1} = \begin{bmatrix} \rho_k \cos \theta_k \\ \rho_k \sin \theta_k \end{bmatrix} + \begin{bmatrix} \cos \theta_k & -\rho_k \sin \theta_k \\ \sin \theta_k & \rho_k \cos \theta_k \end{bmatrix} \left[ \begin{bmatrix} \cos \theta_{k+1} & -\rho_{k+1} \sin \theta_{k+1} \\ \sin \theta_{k+1} & \rho_{k+1} \cos \theta_{k+1} \end{bmatrix} - \begin{bmatrix} \cos \theta_k & -\rho_k \sin \theta_k \\ \sin \theta_k & \rho_k \cos \theta_k \end{bmatrix} \right]$$

## 5 卡尔曼滤波应用

在机器人基于测距仪的跟踪中，激光测距仪的采样周期为 100ms，假设在  $k$  时刻获取了实测值并结合  $k-1$  时刻的距离预测值（方程 1），则可以根据方程 4 对  $k$  时刻的测量值进行滤波估计修正。修正之后就可以对  $k+1$  时刻的距离进行预测，相当于提前了一个采样周期。因而可以提前进行运动控制。



以机器人上的激光测距仪检测跟踪动态目标为例，说明卡尔曼滤波在滤波和预测上的应用。

考虑机器人的运动状态，可以分三种情况分别进行机器人坐标系下的障碍目标状态估计：1 机器人静止而目标运动；2 机器人运动而目标静止，相当于机器人对静止的环境或障碍、目标进行相对位置的估计，有对机器人定位修正的作用。（因为机器人运动时通过里程计等确定的相对运动位置会有误差，需要校订；另外相邻两个运动周期和激光采样周期内机器人与目标的相对位置只通过激光测距仪来确定存在误差，结合里程计对机器人位置的计算可以通过卡尔曼滤波更加准确估计二者的相对位置。）3 当机器人和目标都处于运动状态时，预测运动目标相对于机器人的相对运动，从而进行机器人控制。当然第三种情况可以包含前两种，如果能够通过在第三种情况的算法中改变不同的参数得到前两种情况的计算，则就可以用一套方法对动态环境中的动、静目标进行相对运动预测了。

目前认为机器人是静止的，机器人的传感器坐标采用极坐标系。

设系统的状态向量  $X(k) = [x \quad \dot{x} \quad y \quad \dot{y}]^T$ ,

观测系统的状态方程为（没有输入量，所以  $B$  为 0）： $X(k) = AX(k-1) + W(k-1)$ 。

由于激光测距仪的采样周期  $t$  为 100ms，而机器人跟踪的目标的运动速度比较平缓，所以在局部时间段内设定目标为匀速运动，所以

$$x(k) = x(k-1) + \dot{x}(k-1)t, \quad (6)$$

$t$  为采样周期。

而沿  $x$  方向的运动速度是变化的，可以取前一个采样周期内速度的均值（或者前两个周期内的速度均值）。

$$\dot{x}(k-1) = \frac{x(k-1|k-1) - x(k-2|k-2)}{t}, \text{ 而 } \dot{x}(k) = \frac{x(k|k-1) - x(k-1|k-1)}{t}$$

$$\text{所以 } \dot{x}(k) = \frac{x(k|k-1) - x(k-1|k-1)}{x(k-1|k-1) - x(k-2|k-2)} \dot{x}(k-1), \text{ 令 } a(k) = \frac{x(k|k-1) - x(k-1|k-1)}{x(k-1|k-1) - x(k-2|k-2)}, \text{ 则}$$

$$\dot{x}(k) = a(k)\dot{x}(k-1) \quad (7)$$

同理可以得到  $y$  方向的运动方程：

$$y(k) = y(k-1) + \dot{y}(k-1)t \quad (8)$$

$$\dot{y}(k) = b(k)\dot{y}(k-1) \quad (9)$$

$$\text{式中： } b(k) = \frac{y(k|k-1) - y(k-1|k-1)}{y(k-1|k-1) - y(k-2|k-2)}$$

由（6）—（9）式得到系统的状态方程：

$$X(k) = AX(k-1) + W(k-1)$$

$$\text{式中： } A = \begin{bmatrix} 1 & t & 0 & 0 \\ 0 & a(k) & 0 & 0 \\ 0 & 0 & 1 & t \\ 0 & 0 & 0 & b(k) \end{bmatrix}, \quad W(k-1) \text{ 为过程噪声，是均值为零的高斯白噪声，协方差为 } Q(k-1)。$$

如果过程噪声  $W(k-1)$  用  $T(k-1)v(k-1)$  来代替,  $T(k)$  为过程噪声的分布矩阵,  $v(k)$  为各状态变量对应的误差向量, 由于各状态变量的误差符合 0 均值的高斯分布 (0 均值的正态分布), 即符合  $N(0, \sigma^2)$ , 则  $Q(k)$  变为  $T(k)\sigma_v^2(k)T^T(k)$  (见《雷达数据处理及应用》-21)

测量方程为:  $Z(k) = CX(k) + V(k)$ ,

式中:  $Z(k) = \begin{bmatrix} z_1(k) \\ z_2(k) \end{bmatrix} = \begin{bmatrix} x(k) \\ y(k) \end{bmatrix} = \begin{bmatrix} \rho \cos \theta \\ \rho \sin \theta \end{bmatrix}$ ,  $\rho$  和  $\theta$  分别为极坐标下激光雷达的目标径向距离和方位角

的测量数据。

在上面的过程方程中, 以前后两个采样周期的位置差来求速度, 进行考虑目标加速运行的状态存在一定的不足。因为每次的估计的位置仍然存在一定的误差, 那么两次的位置估计值相减, 有可能减小总体的误差, 也可能放大总体误差 (即相减之后的误差), 所以可能会造成速度的振荡变化, 即使目标匀速运动也可能使滤波出现跟踪中的波动, 降低跟踪及预测效果。

### 5.1 雷达跟踪

见《数字滤波和卡尔曼滤波—1982》中第 8、9 章中的部分内容。