## 1 代码对应的公式



公式1.1 先验估计预测

说明：从k-1时刻的最优估计值转移到k时刻的先验估计值，这个公式承接了两个时刻的值的转移。在利用前一时刻的最优估计的最小二乘中有用到。



公式1.2 先验误差协方差矩阵预测

说明：协方差的先验估计，理解起来和公式1.1一致。



公式1.3 卡尔曼增益



公式1.4 X校正公式

说明：该增益值用来解释k时刻的估计值更偏重于到底是相信当前时刻的先验估计还是当前时刻的测量值。 极端情况Kk=0 或者1.



公式1.5 误差协方差矩阵

说明：误差协方差矩阵的更新公式。误差协方差矩阵就是关于估计误差的一个矩阵，因为状态有两个，位移和速度，所以协方差矩阵是一个2x2的矩阵。



公式1.6 kalman估计误差



公式1.7 样本方差

其中是误差均值.因为该实验是100个时刻，每个时刻又重复做了很多次，所以可以根据样本方差来算出真实的误差方差，比如，系统状态方程的Wk噪声。，因为期望为0，所以该协方差矩阵就是对角矩阵。



公式1.8

## 2 思路

vector是一个1000x100的矩阵，1000是每个时刻重复的试验次数，100是采样时间，由于模拟的是无运动噪声的系统，所以只存在观测噪声。

Vector矩阵中记录每次实验每个时刻的位移误差，在模拟完1000次的试验之后，对vector矩阵中的每一列求方差，并将100个时刻的方差记录在Difference行向量中。

同时，将每一次实验的误差估计协方差存储在errPx矩阵中。将Difference和errPx进行比较即可。

## 3 源代码

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%function main3\_3\_2

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

N=100; %仿真时间，时间序列总数

vector=zeros(1000,100); % 位移的误差

vector1=zeros(1000,100); % 速度的误差

for kk=1:1000

% 定义一个行向量存储每次实验每个时刻的方差

% 噪声

Q=[0,0;0,0]; % 过程噪声方差为0，即下落过程忽略空气阻力

R=1; % 观测噪声方差

W=sqrt(Q)\*randn(2,N);% 既然Q为0，则W=0；在此写出，方便对照理解

V=sqrt(R)\*randn(1,N);% 测量噪声V(k)

% 系统矩阵

A=[1,1;0,1]; %状态转移矩阵

B=[0.5;1]; %控制量

U=-1;

H=[1,0]; %观测矩阵

% 初始化

X=zeros(2,N); % 物体真实状态

X(:,1)=[95;1]; % 初始位移和速度

P0=[10,0;0,1]; % 初始误差

Z=zeros(1,N);

Z(1)=H\*X(:,1); % 初始观测值

Xkf=zeros(2,N);% Kalman估计状态初始化

Xkf(:,1)=X(:,1)+sqrt(P0)\*randn(2,1); % 在初始化的时候加入误差方差估计值 从而保持吻合

err\_P=zeros(N,2);

err\_P(1,1)=P0(1,1);

err\_P(1,2)=P0(2,2);

I=eye(2); % 二维系统

% 第一列

vector(kk,1)=Xkf(1,1)-X(1,1);

vector1(kk,1)=Xkf(2,1)-X(2,1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=2:N

%物体下落，受状态方程的驱动

X(:,k)=A\*X(:,k-1)+B\*U+W(k);

% 位移传感器对目标进行观测

Z(k)=H\*X(:,k)+V(k);

% Kalman滤波

X\_pre=A\*Xkf(:,k-1)+B\*U; %状态预测

P\_pre=A\*P0\*A'+Q; %协方差预测

Kg=P\_pre\*H'\*inv(H\*P\_pre\*H'+R); %计算Kalman增益

Xkf(:,k)=X\_pre+Kg\*(Z(k)-H\*X\_pre); % 状态更新

P0=(I-Kg\*H)\*P\_pre;%方差更新

vector(kk,k)=Xkf(1,k)-X(1,k); % 第kk次实验的第k个时刻 位移的误差

vector1(kk,k)=Xkf(2,k)-X(2,k); % 第kk次实验的第k个时刻 速度的误差

% 误差均方值

err\_P(k,1)=P0(1,1);

err\_P(k,2)=P0(2,2);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

end

% 计算vector矩阵中 每一列的方差 存储在行向量中

Difference=var(vector,0,1);

Difference1=var(vector1,0,1);

errPx=transpose(err\_P);

errPx1=errPx(1,:);

errPx2=errPx(2,:);

% 将err\_P(k,1)和Difference()画在同一张图中，比较卡尔曼滤波估计出来的Pk与实际的方差相差多少.

figure

plot(Difference,'-bo'); % 实际的位移误差方差

hold on;

plot(errPx1,'-g+');

legend('实际位移误差方差','kalman估计方差');

xlabel('采样时间/s');

ylabel('方差');

figure

plot(Difference1,'-bo'); %实际的速度误差方差

hold on;

plot(errPx2,'-g+');

legend('实际速度误差方差','kalman估计方差');

xlabel('采样时间/s');

ylabel('方差');

## 4 实验结果

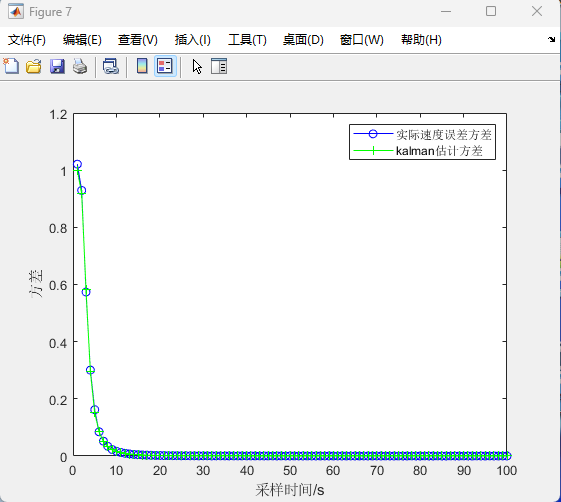


图3.1 实际速度误差方差和kalman估计速度方差比较

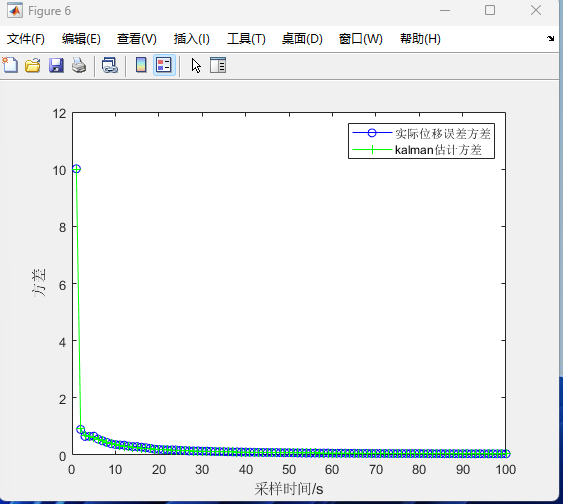


图3.2 实际位移误差方差和kalman估计误差方差比较