

最优估计实验报告

姓名： 秦旗峰

学号： 2023302143029

专业： 导航工程（智能导航实验班）

课程： 最优估计 2

武汉大学

2025.05

目录

1 实验任务.....	3
2 算法原理.....	3
2.1 扩展 Kalman 滤波.....	3
2.1.1 时间预测.....	3
2.1.2 测量更新.....	4
2.2 观测值逐次更新的扩展 Kalman 滤波.....	5
3 物理建模.....	7
3.1 问题描述.....	7
3.2 模型应用.....	8
3.2.1 时间预测.....	8
3.2.2 EKF 测量更新.....	10
3.2.3 观测值逐次更新的 EKF 测量更新.....	10
4 编程实现.....	12
4.1 结构体.....	12
4.2 解算函数.....	12
4.3 主函数.....	13
4.4 结果可视化.....	13
5 结果分析.....	14
6 总结与反思.....	19

1 实验任务

- **主要任务：**利用扩展 Kalman 滤波和观测值逐次更新的扩展 Kalman 滤波对平抛运动物体轨迹进行滤波估计，实现编程解算和可视化分析。
- **实验目的：**加深对扩展 Kalman 滤波的理解，掌握编程解算扩展 Kalman 滤波的实现步骤，通过分析滤波结果，找到扩展 Kalman 滤波的普遍规律。

2 算法原理

2.1 扩展 Kalman 滤波

Kalman 滤波是基于线性模型的估计，但在实际应用中，描述运动状态的微分方程和观测方程大多是非线性的。非线性方程线性化过程往往会因为初值偏差较大带来较大的线性化误差，为了减小模型线性化带来的模型误差，扩展的 Kalman 滤波(EKF)在 t_{k-1} 到 t_k 时间段预测中，将 $\hat{\mathbf{X}}(k-1)$ 作为近似值线性化微分方程；在 t_k 的测量更新时，将 $\hat{\mathbf{X}}(k, k-1)$ 作为近似值线性化观测方程，测量更新得到 $\hat{\mathbf{X}}(k)$ 。

2.1.1 时间预测

设非线性连续时间系统的状态方程为：

$$\dot{\mathbf{X}}(t) = \mathbf{g}[\mathbf{X}(t), \mathbf{u}(t), \mathbf{e}(t)] \quad (1)$$

首先将式(1)线性化。在 (t_{k-1}, t_k) 时间段的预测中， $\hat{\mathbf{X}}(k-1)$ 为近似值，线性化后得到：

$$\dot{\mathbf{X}}(t) = \mathbf{A}(t)\mathbf{X}(t) + \mathbf{G}(t) + \mathbf{C}(t)\mathbf{e}(t) \quad (2)$$

其中：

$$\begin{aligned} \mathbf{A}(t) &= \left[\frac{\partial \mathbf{g}}{\partial \mathbf{X}(t)} \right]_{\mathbf{X}(t)=\hat{\mathbf{X}}(k-1)} \\ \mathbf{C}(t) &= \left[\frac{\partial \mathbf{g}}{\partial \mathbf{e}(t)} \right]_{\mathbf{X}(t)=\hat{\mathbf{X}}(k-1)} \\ \mathbf{G}(t) &= \mathbf{g}[\hat{\mathbf{X}}(k-1), \mathbf{u}(t), \mathbf{0}] - \mathbf{A}(t)\hat{\mathbf{X}}(k-1) \end{aligned} \quad (3)$$

再解式(2)的微分方程，得到离散化后的状态方程：

$$\mathbf{X}(k) = \Phi_{k,k-1}\mathbf{X}(k-1) + \Omega(k-1) + \mathbf{w}(k-1) \quad (4)$$

其中：

$$\begin{aligned} \Phi_{k,k-1} &= e^{\int_{t_{k-1}}^{t_k} A(\tau) d\tau} \\ \Omega(k-1) &= \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) G(\tau) d\tau \\ \mathbf{w}(k-1) &= \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) C(\tau) \mathbf{e}(\tau) d\tau \end{aligned} \quad (5)$$

基于(4)给出的状态方程，时间预测为：

$$\hat{\mathbf{X}}(k, k-1) = \Phi_{k,k-1}\hat{\mathbf{X}}(k-1) + \Omega(k-1) \quad (6)$$

$\hat{\mathbf{X}}(k, k-1)$ 的方差为：

$$\mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) = \Phi_{k,k-1}\mathbf{D}_{\hat{\mathbf{X}}(k-1)}\Phi_{k,k-1}^T + \mathbf{D}_{\mathbf{w}(k-1)} \quad (7)$$

其中：

$$\mathbf{D}_{\mathbf{w}(k-1)} = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) C(\tau) \mathbf{D}_e(\tau) C^T(\tau) \Phi^T(t_k, \tau) d\tau \quad (8)$$

2.1.2 测量更新

对 t_k 时刻 $\mathbf{X}(k)$ 的观测方程为：

$$\mathbf{Z}(k) = \mathbf{F}[\mathbf{X}(k)] + \Delta(k) \quad (9)$$

在测量更新时将观测方程在近似值 $\mathbf{X}^*(k)$ 处用泰勒公式展开并舍去高阶项得到：

$$\mathbf{Z}(k) = \left[\frac{\partial \mathbf{F}}{\partial \mathbf{X}(k)} \right]_{\mathbf{X}(k)=\mathbf{X}^*(k)} (\mathbf{X}(k) - \mathbf{X}^*(k)) + \mathbf{F}[\mathbf{X}^*(k)] + \Delta(k) \quad (10)$$

令：

$$\left[\frac{\partial \mathbf{F}}{\partial \mathbf{X}(k)} \right]_{\mathbf{X}(k)=\mathbf{X}^*(k)} = \mathbf{H}_k \quad (11)$$

则线性化后的观测方程为：

$$\mathbf{Z}(k) = \mathbf{H}_k \mathbf{X}(k) + \{\mathbf{F}[\mathbf{X}^*(k)] - \mathbf{H}_k \mathbf{X}^*(k)\} + \Delta(k) \quad (12)$$

其中 $\{\mathbf{F}[\mathbf{X}^*(k)] - \mathbf{H}_k \mathbf{X}^*(k)\}$ 为非随机部分。测量更新的新息为：

$$\mathbf{V}(k) = \mathbf{Z}(k) - \{\mathbf{H}_k \hat{\mathbf{X}}(k, k-1) + \mathbf{F}[\mathbf{X}^*(k)] - \mathbf{H}_k \mathbf{X}^*(k)\} \quad (13)$$

由于 $\hat{\mathbf{X}}(k, k-1)$ 是在 t_k 时刻无观测值时的最优估计，所以取 $\mathbf{X}^*(k) = \hat{\mathbf{X}}(k, k-1)$ ，那么式(13)可以写为：

$$\mathbf{V}(k, k-1) = \mathbf{Z}(k) - \mathbf{F}[\hat{\mathbf{X}}(k, k-1)] \quad (14)$$

在得到新息后，测量更新为：

$$\hat{\mathbf{X}}(k) = \hat{\mathbf{X}}(k, k-1) + \mathbf{K}_k \mathbf{V}(k, k-1) \quad (15)$$

$$\mathbf{K}_k = \mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) \mathbf{H}_k^T \left(\mathbf{H}_k \mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) \mathbf{H}_k^T + \mathbf{D}_{\Delta}(k) \right)^{-1} \quad (16)$$

其方差为：

$$\mathbf{D}_{\hat{\mathbf{X}}}(k) = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) \quad (17)$$

2.2 观测值逐次更新的扩展 Kalman 滤波

观测值逐次更新的扩展 Kalman 滤波算法中的时间预测与 2.1 节中扩展 Kalman 滤波时间预测相同。只是在做观测值更新时将观测值向量拆分，用每一个观测值逐一对状态进行更新。

设有观测方程：

$$\mathbf{Z}(k) = \mathbf{F}[\mathbf{X}(k)] + \Delta(k) \quad (18)$$

将观测方程展开为：

$$\begin{bmatrix} Z_1(k) \\ Z_2(k) \\ \vdots \\ Z_l(k) \end{bmatrix} = \begin{bmatrix} f_1(\mathbf{X}(k)) \\ f_2(\mathbf{X}(k)) \\ \vdots \\ f_l(\mathbf{X}(k)) \end{bmatrix} + \begin{bmatrix} \Delta_1(k) \\ \Delta_2(k) \\ \vdots \\ \Delta_l(k) \end{bmatrix} \quad (19)$$

若观测值之间不相关，量测噪声矩阵为对角矩阵：

$$\mathbf{D}_{\Delta}(k) = \text{diag}[d_1(k), d_2(k), \dots, d_l(k)] \quad (20)$$

在测量更新时，首先取第一个观测方程：

$$Z_1(k) = f_1(\mathbf{X}(k)) + \Delta_1(k) \quad (21)$$

将观测方程线性化，取 $\mathbf{X}^*(k) = \hat{\mathbf{X}}(k, k-1)$ ，得到雅各布矩阵：

$$\left[\frac{\partial f_1}{\partial \mathbf{X}(k)} \right]_{\mathbf{X}(k)=\hat{\mathbf{X}}(k, k-1)} = \mathbf{h}_1(k) \quad (22)$$

和 $Z_1(k)$ 的新息：

$$V_1(k, k-1) = Z_1(k) - f_1(\hat{\mathbf{X}}(k, k-1)) \quad (23)$$

对时间预测的更新为：

$$\mathbf{K}_k^{[1]} = \frac{\mathbf{D}_{\hat{\mathbf{X}}}(k, k-1)\mathbf{h}_1(k)^T}{\mathbf{h}_1(k)\mathbf{D}_{\hat{\mathbf{X}}}(k, k-1)\mathbf{h}_1(k)^T + d_1(k)} \quad (24)$$

$$\hat{\mathbf{X}}^{[1]}(k) = \hat{\mathbf{X}}(k, k-1) + \mathbf{K}_k^{[1]}V_1(k, k-1) \quad (25)$$

$$\mathbf{D}_{\hat{\mathbf{X}}}^{[1]}(k) = \left(\mathbf{I} - \mathbf{K}_k^{[1]}\mathbf{h}_1(k)\right)\mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) \quad (26)$$

接着，取 $\mathbf{X}^*(k) = \hat{\mathbf{X}}^{[1]}(k)$ 对第二个观测方程进行线性化并对预测值更新，得到 $\hat{\mathbf{X}}^{[2]}(k)$ 和 $\mathbf{D}_{\hat{\mathbf{X}}}^{[2]}(k)$ ，如此进行下去。第 j 个观测值 $Z_j(k)$ 对状态 $\hat{\mathbf{X}}^{[j-1]}(k)$ 的更新为：

$$\left[\frac{\partial f_j}{\partial \mathbf{X}(k)}\right]_{\mathbf{X}(k)=\hat{\mathbf{X}}^{[j-1]}(k)} = \mathbf{h}_j(k) \quad (27)$$

$$V_j(k, k-1) = Z_j(k) - f_j\left(\hat{\mathbf{X}}^{[j-1]}(k)\right) \quad (28)$$

$$\mathbf{K}_k^{[j]} = \frac{\mathbf{D}_{\hat{\mathbf{X}}}^{[j-1]}(k)\mathbf{h}_j(k)^T}{\mathbf{h}_j(k)\mathbf{D}_{\hat{\mathbf{X}}}^{[j-1]}(k)\mathbf{h}_j(k)^T + d_j(k)} \quad (29)$$

$$\hat{\mathbf{X}}^{[j]}(k) = \hat{\mathbf{X}}^{[j-1]}(k) + \mathbf{K}_k^{[j]}V_j(k, k-1) \quad (30)$$

$$\mathbf{D}_{\hat{\mathbf{X}}}^{[j]}(k) = \left(\mathbf{I} - \mathbf{K}_k^{[j]}\mathbf{h}_j(k)\right)\mathbf{D}_{\hat{\mathbf{X}}}^{[j-1]}(k) \quad (31)$$

上式中，当 $j = 1$ ，即在第一个观测值进行更新时：

$$\hat{\mathbf{X}}^{[1]}(k) = \hat{\mathbf{X}}(k, k-1) \quad (32)$$

$$\mathbf{D}_{\hat{\mathbf{X}}}^{[1]}(k) = \mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) \quad (33)$$

重复以上步骤，直到观测向量 $\mathbf{Z}(k)$ 中最后一个观测值 $Z_l(k)$ 对状态更新完毕，得到 $\hat{\mathbf{X}}^{[l]}(k)$ 和 $\mathbf{D}_{\hat{\mathbf{X}}}^{[l]}(k)$ 。 $\hat{\mathbf{X}}^{[l]}(k)$ 和 $\mathbf{D}_{\hat{\mathbf{X}}}^{[l]}(k)$ 即为在 k 时刻所有观测值对时间预测 $\hat{\mathbf{X}}(k, k-1)$ 的更新。接下来进行 $k+1$ 时刻的滤波计算，先进行时间预测，再对时间预测进行观测值的逐次更新。

若观测值之间存在相关性，则先对观测方程去相关操作，再进行滤波处理。

3 物理建模

3.1 问题描述

一个从空中水平抛射出的物体，初始水平速度为 $v_x(0)$,初始位置坐标 $(x(0),y(0))$ ；受重力 g 和阻尼力的影响，阻尼力与速度平方成正比，水平和垂直阻尼系数分别为 k_x, k_y 。此外，还存在不确定干扰力，沿 x, y 轴分别为 e_x, e_y 。利用一设备可以观测该物体得到距离 r 和角度 α 。

已知： $k_x = 0.01, k_y = 0.05$ ，重力加速度 $g = 9.8m/s^2$ ；初始位置和速度及其方差为：

$$\mathbf{X}(t_0) = \begin{bmatrix} x(t_0) \\ v_x(t_0) \\ y(t_0) \\ v_y(t_0) \end{bmatrix} = \begin{bmatrix} 0m \\ 50m \\ \frac{s}{500m} \\ 0m \\ \frac{s}{500m} \end{bmatrix} \quad (34)$$

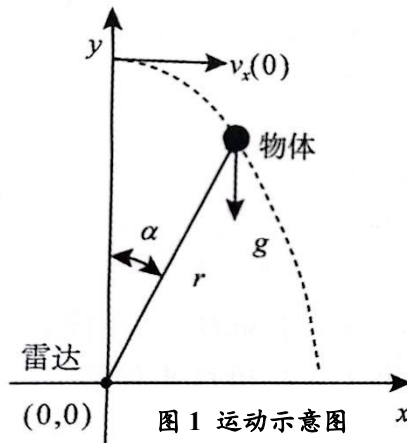
$$\mathbf{D}_{\mathbf{X}}(t_0) = \begin{bmatrix} 100m^2 & & & \\ & 100(m/s)^2 & & \\ & & 100m^2 & \\ & & & 100(m/s)^2 \end{bmatrix} \quad (35)$$

将干扰力 e_x 和 e_y 视为零均值白噪声，并且：

$$\mathbf{e}(t) = \begin{bmatrix} e_x \\ e_y \end{bmatrix}, Cov[\mathbf{e}(t), \mathbf{e}(\tau)] = \mathbf{D}_e(t)\delta(t - \tau), \mathbf{D}_e(t) = \begin{bmatrix} 1.5^2 & 0 \\ 0 & 1.5^2 \end{bmatrix} \left(\frac{m^2}{s^2} \right) \quad (36)$$

观测值采样间隔为 $\Delta t = 0.1s$ ，观测噪声与系统噪声不相关，并且：

$$\Delta(k) = \begin{bmatrix} \Delta_r(k) \\ \Delta_\alpha(k) \end{bmatrix}, Cov[\Delta(k), \Delta(j)] = \mathbf{D}_\Delta(k)\delta(k - j) \quad (37)$$



$$\mathbf{D}_{\Delta}(k) = \begin{bmatrix} 100m^2 & 0 \\ 0 & 0.0001rad^2 \end{bmatrix} \quad (38)$$

3.2 模型应用

3.2.1 时间预测

设状态为 $\mathbf{X}(t) = [x(t) \ v_x(t) \ y(t) \ v_y(t)]^T$ ，可以列出状态方程为：

$$\mathbf{g}: \begin{cases} \dot{x}(t) = v_x(t) \\ \dot{v}_x(t) = -k_x v_x^2 + e_x(t) \\ \dot{y}(t) = v_y(t) \\ \dot{v}_y(t) = k_y v_y^2 - g + e_y(t) \end{cases} \quad (39)$$

由于式(39)表示的微分方程为非线性，先将其线性化。在 $\mathbf{X}(k) = \mathbf{X}^*(k)$ ， $\mathbf{e}(t) = \mathbf{e}^*(t) = \mathbf{0}$ 处线性化，得到：

$$\dot{\mathbf{X}}(t) = \mathbf{A}(t)\mathbf{X}(t) + \mathbf{g}[\mathbf{X}^*(k) \ \mathbf{u}(t) \ 0] - \mathbf{A}(t)\mathbf{X}^*(t) + \mathbf{C}(t)\mathbf{e}(t) \quad (40)$$

其中：

$$\mathbf{A}(t) = \left[\frac{\partial \mathbf{g}}{\partial \mathbf{X}(t)} \right]^* = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -2k_x v_x(t) & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 2k_y v_y(t) \end{bmatrix}_{\mathbf{X}(k)=\mathbf{X}^*(k)} \quad (41)$$

$$\mathbf{g}[\mathbf{X}^*(k) \ \mathbf{u}(t) \ 0] = \begin{bmatrix} v_x(t) \\ -k_x v_x^2 \\ v_y(t) \\ k_y v_y^2 - g \end{bmatrix}_{\mathbf{X}(k)=\mathbf{X}^*(k)} \quad (42)$$

$$\mathbf{C}(t) = \left[\frac{\partial \mathbf{g}}{\partial \mathbf{e}(t)} \right]^* = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \quad (43)$$

令：

$$\mathbf{G}(t) = \mathbf{g}[\mathbf{X}^*(k) \ \mathbf{u}(t) \ 0] - \mathbf{A}(t)\mathbf{X}^*(t) \quad (44)$$

$$\mathbf{G}(t) = \begin{bmatrix} 0 \\ k_x v_x^2 \\ 0 \\ -k_y v_y^2 - g \end{bmatrix}_{\mathbf{X}(k)=\mathbf{X}^*(k)} \quad (45)$$

注意到在一次时间预测内， $\mathbf{A}(t)$ 、 $\mathbf{C}(t)$ 和 $\mathbf{G}(t)$ 在代入 $\mathbf{X}^*(k)$ 后均为常矩阵，因此下面的推到中用 \mathbf{A} 、 \mathbf{C} 和 \mathbf{G} 分别代替 $\mathbf{A}(t)$ 、 $\mathbf{C}(t)$ 和 $\mathbf{G}(t)$ 。

状态转移矩阵由式(5)展开可得：

$$\Phi(t, \tau) = I + A \times (t - \tau) \quad (46)$$

$$\Phi(t, \tau) = \begin{bmatrix} 1 & t - \tau & 0 & 0 \\ 0 & 1 - 2k_x v_x^* \times (t - \tau) & 0 & 0 \\ 0 & 0 & 1 & t - \tau \\ 0 & 0 & 0 & 1 + 2k_y v_y^* \times (t - \tau) \end{bmatrix} \quad (47)$$

由于采样间隔为 $\Delta t = 0.1s$ ，所以时间 t_{k-1} 到时间 t_k 的状态转移矩阵为：

$$\Phi_{k,k-1} = \begin{bmatrix} 1 & 0.1 & 0 & 0 \\ 0 & 1 - 2k_x v_x^* \times 0.1 & 0 & 0 \\ 0 & 0 & 1 & 0.1 \\ 0 & 0 & 0 & 1 + 2k_y v_y^* \times 0.1 \end{bmatrix} \quad (48)$$

离散化后的状态方程：

$$X(k) = \Phi_{k,k-1} X(k-1) + \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) d\tau \cdot G + \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) C e(\tau) d\tau \quad (49)$$

令 $\Omega(k-1) = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) d\tau G$ ， $w(k-1) = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) C e(\tau) d\tau$ ，可得：

$$\begin{aligned} \Omega(k-1) &= \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) d\tau \cdot G \\ &= \begin{bmatrix} 0.1 & 0.005 & 0 & 0 \\ 0 & 0.1 - k_x v_x^* \times 0.01 & 0 & 0 \\ 0 & 0 & 0.1 & 0.05 \\ 0 & 0 & 0 & 0.1 + k_y v_y^* \times 0.01 \end{bmatrix} \begin{bmatrix} 0 \\ k_x v_x^{*2} \\ 0 \\ -k_y v_y^{*2} - g \end{bmatrix} \end{aligned} \quad (50)$$

$w(k-1)$ 的方差为：

$$\begin{aligned} D_{w(k-1)} &= \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) C D_e(\tau) C^T \Phi^T(t_k, \tau) d\tau = \\ &= \begin{bmatrix} \frac{2.25}{3} \times 0.001 & \left(0.005 - \frac{0.002 \times k_x v_x^*}{3}\right) 2.25 & 0 & 0 \\ \left(0.005 - \frac{0.002 \times k_x v_x^*}{3}\right) 2.25 & \frac{0.3 - 0.06 k_x v_x^* + 0.004 (k_x v_x^*)^2}{3} \times 2.25 & 0 & 0 \\ 0 & 0 & \frac{0.001}{3} \times 2.25 & \left(0.005 + \frac{0.002 \times k_y v_y^*}{3}\right) \times 2.25 \\ 0 & 0 & \left(0.005 + \frac{0.002 \times k_y v_y^*}{3}\right) \times 2.25 & \frac{0.3 + 0.06 k_y v_y^* + 0.004 (k_y v_y^*)^2}{3} \times 2.25 \end{bmatrix} \end{aligned} \quad (51)$$

在以上模型中：

$$X^*(k) = \begin{bmatrix} x^* \\ v_x^* \\ y^* \\ v_y^* \end{bmatrix} = \begin{bmatrix} \hat{x}(k-1) \\ \hat{v}_x(k-1) \\ \hat{y}(k-1) \\ \hat{v}_y(k-1) \end{bmatrix} \quad (52)$$

因此，时间预测为：

$$\hat{X}(k, k-1) = \Phi_{k,k-1} \hat{X}(k-1) + \Omega(k-1) \quad (53)$$

其方差为:

$$\mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) = \mathbf{\Phi}_{k,k-1} \mathbf{D}_{\hat{\mathbf{X}}(k-1)} \mathbf{\Phi}_{k,k-1}^T + \mathbf{D}_{\mathbf{w}(k-1)} \quad (54)$$

3.2.2 EKF 测量更新

对于扩展 Kalman 滤波，观测方程为:

$$\mathbf{Z}(k) = \begin{bmatrix} r(k) \\ \alpha(k) \end{bmatrix} = \begin{bmatrix} \sqrt{x^2(k) + y^2(k)} \\ \arctan \frac{x(k)}{y(k)} \end{bmatrix} + \begin{bmatrix} \Delta_r(k) \\ \Delta_\alpha(k) \end{bmatrix} \quad (55)$$

对时间预测的更新的新息为:

$$\mathbf{V}(k, k-1) = \begin{bmatrix} r(k) \\ \alpha(k) \end{bmatrix} - \begin{bmatrix} \sqrt{x^{*2} + y^{*2}} \\ \arctan \frac{x^*}{y^*} \end{bmatrix} \quad (56)$$

$$\mathbf{H}_k = \left[\frac{\partial \mathbf{Z}}{\partial \mathbf{X}(k)} \right]_{\mathbf{X}(k)=\mathbf{X}^*(k)} = \begin{bmatrix} \frac{x^*}{\sqrt{x^{*2} + y^{*2}}} & 0 & \frac{y^*}{\sqrt{x^{*2} + y^{*2}}} & 0 \\ \frac{y^*}{x^{*2} + y^{*2}} & 0 & -\frac{x^*}{y^{*2} - x^{*2}} & 0 \end{bmatrix} \quad (56)$$

增益矩阵为:

$$\mathbf{K}_k = \mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) \mathbf{H}_k^T \left(\mathbf{H}_k \mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) \mathbf{H}_k^T + \mathbf{D}_{\Delta}(k) \right)^{-1} \quad (57)$$

测量更新为:

$$\hat{\mathbf{X}}(k) = \hat{\mathbf{X}}(k, k-1) + \mathbf{K}_k \mathbf{V}(k, k-1) \quad (58)$$

$$\mathbf{D}_{\hat{\mathbf{X}}}(k) = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) \quad (59)$$

3.2.3 观测值逐次更新的 EKF 测量更新

对于观测值逐次更新的扩展 Kalman 滤波，观测方程为:

$$\begin{cases} f_1: r(k) = \sqrt{x^2(k) + y^2(k)} + \Delta_r(k) \\ f_2: \alpha(k) = \arctan \frac{x(k)}{y(k)} + \Delta_\alpha(k) \end{cases} \quad (55)$$

先利用观测值 $r(k)$ 对时间预测 $\hat{\mathbf{X}}(k, k-1)$ 进行更新:

$$V_1(k, k-1) = r(k) - \sqrt{x^{*2}(k) + y^{*2}(k)} \quad (56)$$

$$\mathbf{h}_1(k) = \left[\frac{\partial f_1}{\partial \mathbf{X}(k)} \right]_{\mathbf{X}(k)=\mathbf{X}^*(k)} = \begin{bmatrix} \frac{x^*}{\sqrt{x^{*2} + y^{*2}}} & 0 & \frac{y^*}{\sqrt{x^{*2} + y^{*2}}} & 0 \end{bmatrix} \quad (57)$$

$$\mathbf{K}_k^{[1]} = \frac{\mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) \mathbf{h}_1(k)^T}{\mathbf{h}_1(k) \mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) \mathbf{h}_1(k)^T + d_r(k)} \quad (58)$$

$$\hat{\mathbf{X}}^{[1]}(k) = \hat{\mathbf{X}}(k, k-1) + \mathbf{K}_k^{[1]} V_1(k, k-1) \quad (59)$$

$$\mathbf{D}_{\hat{\mathbf{X}}}^{[1]}(k) = \left(\mathbf{I} - \mathbf{K}_k^{[1]} \mathbf{h}_1(k) \right) \mathbf{D}_{\hat{\mathbf{X}}}(k, k-1) \quad (60)$$

在利用观测值 $r(k)$ 对时间预测的更新时，近似值取 $\mathbf{X}^*(k) = \hat{\mathbf{X}}(k, k-1)$ 。得到 $\hat{\mathbf{X}}^{[1]}(k)$ 和 $\mathbf{D}_{\hat{\mathbf{X}}}^{[1]}(k)$ ，接着用观测值 $\alpha(k)$ 对 $\hat{\mathbf{X}}^{[1]}(k)$ 和 $\mathbf{D}_{\hat{\mathbf{X}}}^{[1]}(k)$ 更新：

$$V_2(k, k-1) = \alpha(k) - \arctan \frac{x^*}{y^*} \quad (61)$$

$$\mathbf{h}_2(k) = \left[\frac{\partial f_2}{\partial \mathbf{X}(k)} \right]_{\mathbf{X}(k)=\mathbf{X}^*(k)} = \begin{bmatrix} \frac{y^*}{x^{*2} + y^{*2}} & 0 & -\frac{x^*}{y^{*2} + x^{*2}} & 0 \end{bmatrix} \quad (62)$$

$$\mathbf{K}_k^{[2]} = \frac{\mathbf{D}_{\hat{\mathbf{X}}}^{[1]}(k) \mathbf{h}_2(k)^T}{\mathbf{h}_2(k) \mathbf{D}_{\hat{\mathbf{X}}}^{[1]}(k) \mathbf{h}_2(k)^T + d_\alpha(k)} \quad (63)$$

$$\hat{\mathbf{X}}^{[2]}(k) = \hat{\mathbf{X}}^{[1]}(k) + \mathbf{K}_k^{[2]} V_2(k, k-1) \quad (64)$$

$$\mathbf{D}_{\hat{\mathbf{X}}}^{[2]}(k) = \left(\mathbf{I} - \mathbf{K}_k^{[2]} \mathbf{h}_2(k) \right) \mathbf{D}_{\hat{\mathbf{X}}}^{[1]}(k) \quad (65)$$

这时使 $\mathbf{X}^*(k) = \hat{\mathbf{X}}^{[1]}(k)$ ，更新后得到的 $\hat{\mathbf{X}}^{[2]}(k)$ 和 $\mathbf{D}_{\hat{\mathbf{X}}}^{[2]}(k)$ 就是在 k 时刻的滤波结果。在得到 $\hat{\mathbf{X}}(k)$ 和 $\mathbf{D}_{\hat{\mathbf{X}}}(k)$ 后即可对 $k+1$ 时刻进行时间预测和测量更新。

重复以上计算，直到得到最后一组观测值的滤波结果。

4 编程实现

4.1 结构体

按照观测值存储，时间预测，扩展卡尔曼滤波和观测值逐次更新的扩展卡尔曼滤波的两次更新，定义了五个结构体：

```
10 struct Obs //观测值结构体
11 {
12     double time; //时间
13     double R; //距离
14     double alpha; //角度
15 }obs[99]; //99个观测值
```

图 2 观测值结构体

```
17 struct timePredict //时间预测结构体
18 {
19     double time; //时间
20     Eigen::MatrixXd Trans; //状态转移矩阵
21     Eigen::MatrixXd X; //状态变量
22     Eigen::MatrixXd D; //协方差矩阵
23     Eigen::MatrixXd Omega; //离散化后状态方程里与时间无关的变量
24     Eigen::MatrixXd Dw; //状态误差方差阵
25     timePredict() //构造函数赋值
26     {
27         time = 0;
28         Trans = Eigen::MatrixXd(4, 4);
29         X = Eigen::MatrixXd(4, 1);
30         D = Eigen::MatrixXd(4, 4);
31         Omega = Eigen::MatrixXd(4, 1);
32         Dw = Eigen::MatrixXd(4, 4);
33     }
34 }tp[100];
35
36 struct EKF //扩展Kalman滤波
37 {
38     double time; //时间
39     Eigen::MatrixXd obs; //观测值
40     Eigen::MatrixXd f; //观测方程
41     Eigen::MatrixXd V; //新息
42     Eigen::MatrixXd h; //线性化矩阵
43     Eigen::MatrixXd K; //增益矩阵
44     EKF() //构造函数赋值
45     {
46         time = 0;
47         obs = Eigen::MatrixXd(2, 1);
48         f = Eigen::MatrixXd(2, 1);
49         V = Eigen::MatrixXd(2, 1);
50         h = Eigen::MatrixXd(2, 4);
51         K = Eigen::MatrixXd(4, 2);
52     }
53 }ekf[99];
```

图 3 时间预测与 EKF 结构体

```
55 struct R_update //观测值r对预测的更新
56 {
57     double time; //时间
58     double V; //新息
59     Eigen::MatrixXd h; //线性化矩阵
60     Eigen::MatrixXd K; //增益矩阵
61     R_update() //构造函数赋值
62     {
63         time = 0;
64         V = 0;
65         h = Eigen::MatrixXd(1, 4);
66         K = Eigen::MatrixXd(4, 1);
67     }
68 }R_up[99];
69
70 struct Angle_update //观测值角度对预测的更新
71 {
72     double time; //时间
73     double V; //新息
74     Eigen::MatrixXd h; //线性化矩阵
75     Eigen::MatrixXd K; //增益矩阵
76     Angle_update()
77     {
78         time = 0;
79         V = 0;
80         h = Eigen::MatrixXd(1, 4);
81         K = Eigen::MatrixXd(4, 1);
82     }
83 }A_up[99];
```

图 4 观测值测量更新结构体

4.2 解算函数

定义了两个函数，一个是 EKF 解算，另一个是观测值逐次更新的 EKF 解算。

```
1. void EKF(string filename) //扩展 Kalman 滤波解算
```

第一个函数是扩展 Kalman 滤波解算函数。输入文件名，自动读取数据并解算。结果输出到文件中。

```
1. void Obs_EKF(string filename) //观测值逐次更新的扩展 Kalman 滤波
```

第二个函数是观测值逐次更新的扩展 Kalman 滤波解算函数，输入与输出和第一个函数相同。

4.3 主函数

```
1. int main()
2. {
3.     string filename = "D:\\HuaweiMoveData\\Users\\HUAWEI\\Desktop\\ob
    servation.txt";
4.     Obs_EKF(filename);
5.     EKF(filename);
6.     return 0;
7. }
```

主函数输入文件存放的路径，调用两个解算函数就能将结果输出到相应结果文件。

4.4 结果可视化

结果可视化绘图语言使用 Matlab，绘制了滤波估计的轨迹图，坐标 x 和 y 的方差图，观测值的新息图以及观测值带来的增益图。代码较冗长，在附件给出。

5 结果分析

解算结果分别保存在"观测值逐次更新的扩展 kalman 滤波.txt"和"扩展 kalman 滤波.txt"中。通过 Matlab 的绘制函数，我们得以更好分析 EKF 的估计效果。

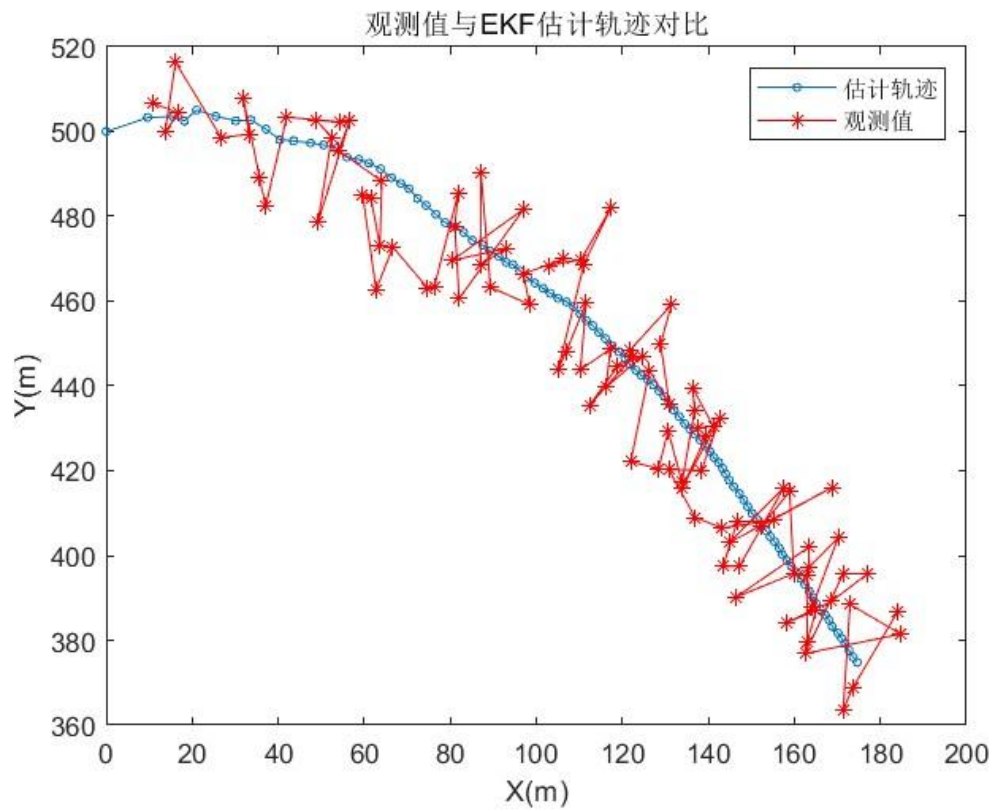


图 5 EKF 估计轨迹与观测值对比

由轨迹图图 5 分析可以得到，观测值轨迹存在较大波动，误差较大。而 EKF 估计轨迹更为平滑，更加符合客观规律。

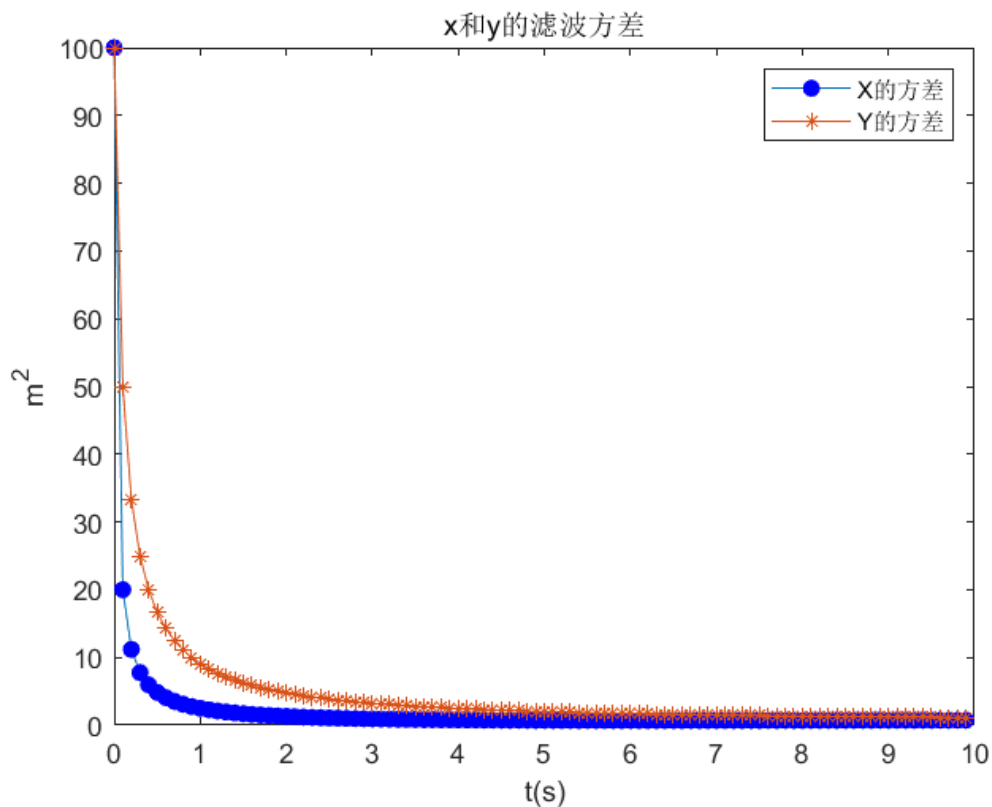


图 6 EKF 滤波方差

x 和 y 坐标的滤波方差图显示了随着滤波的递推，估计方差逐渐减小的过程。 x 和 y 坐标方差由最初的 $100m^2$ 第一次滤波更新就分别下降到约 $20m^2$ 和 $50m^2$ ，收敛速度很快且滤波方差曲线十分平滑，最后收敛到一个较小值。

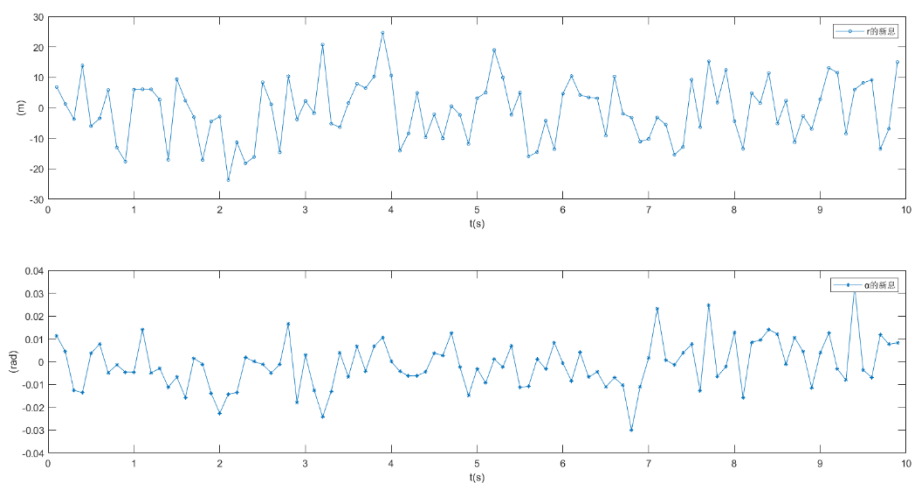


图 7 EKF 观测值新息

观测值新息是观测值与预测值的差异，能够体现观测值噪声大小。不难发现

观测值 r 的新息波动较大，原因是观测值的误差大，观测噪声达到了 $100m^2$ ；而观测值 α 的新息在 $\pm 0.02rad$ 附近波动，误差较小。

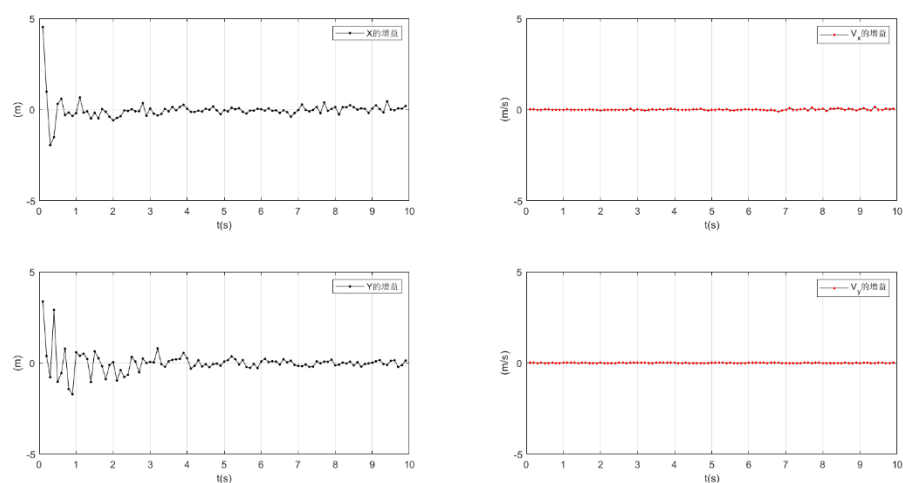


图 8 EKF 观测值对状态的增益

图 8 展示了观测值对时间预测的增益，不难发现，对状态的增益，最终会趋向于 0。由于观测值是坐标的函数，因此对坐标 x 和 y 的增益相较于 v_x 和 v_y 的增益更大。观测值增益波动逐渐减小趋向于零，体现了滤波结果与观测值之间的一致性逐渐增大。

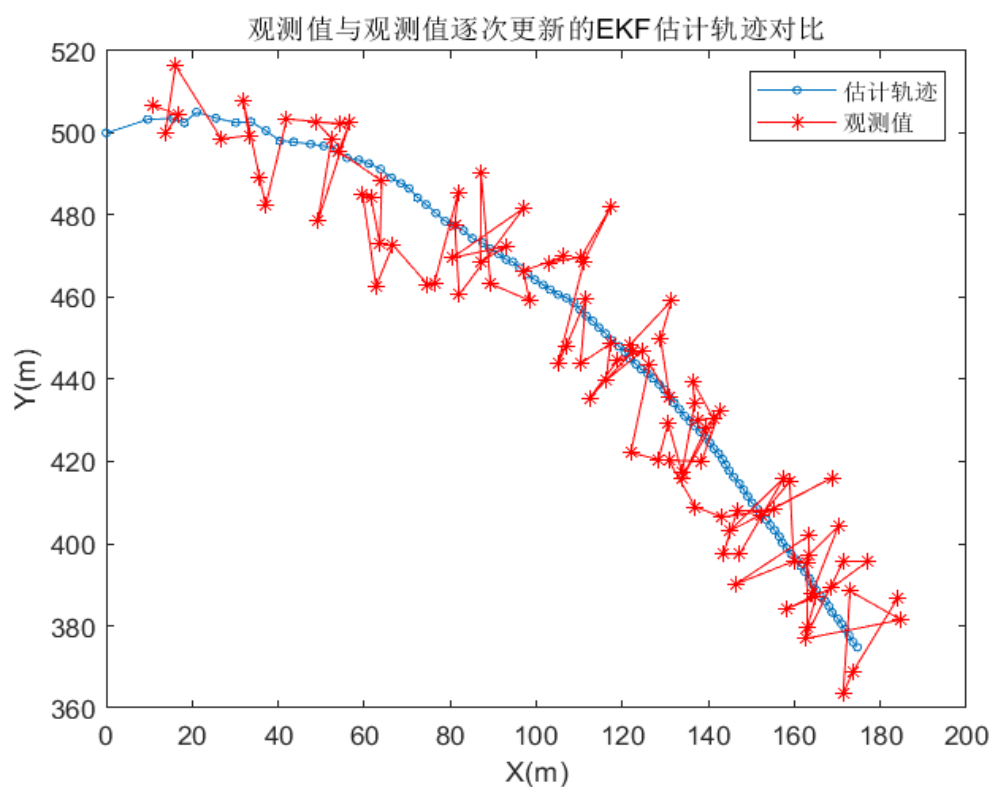


图 9 观测值逐次更新的 EKF 估计轨迹与观测值对比

上图是观测值逐次更新的扩展 Kalman 滤波估计轨迹与观测值的对比图，观测值逐次更新的 EKF 估计轨迹相较于观测值更加平滑并能体现客观规律。观测值在估计轨迹周围波动，也能反映出估计轨迹的正确性。

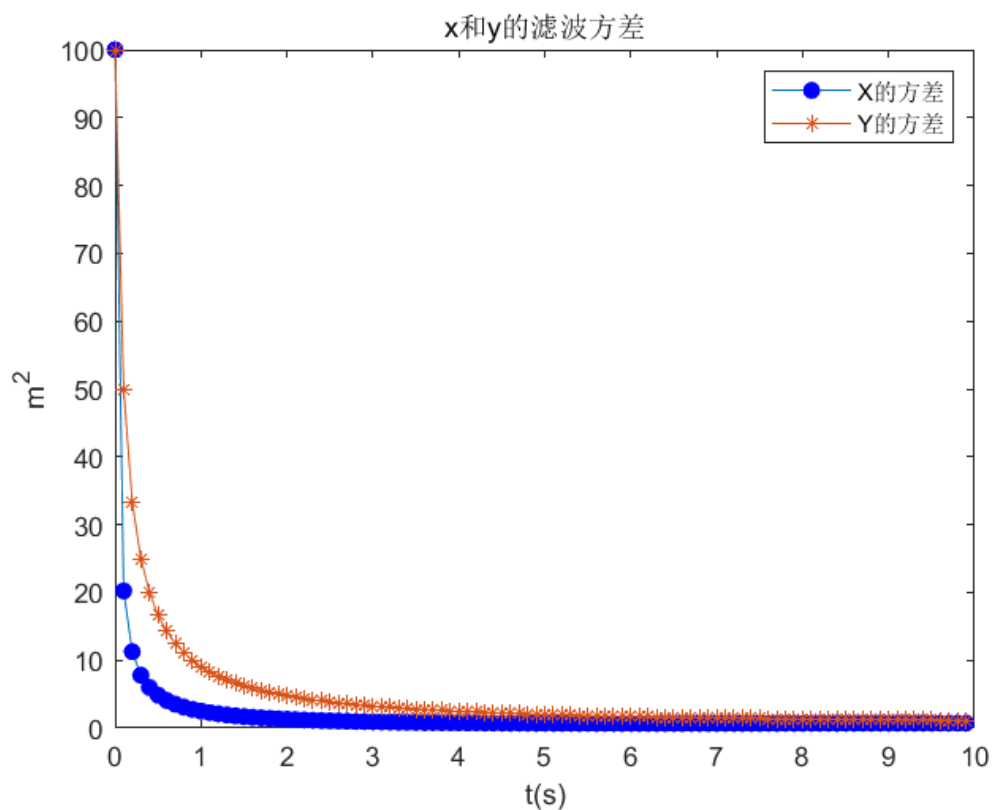


图 10 观测值逐次更新的 EKF 滤波方差

观测值逐次更新的扩展 Kalman 滤波估计的坐标 x 和 y 的方差从最初的 $100m^2$ ，快速收敛到接近于0处，精度显著提高。

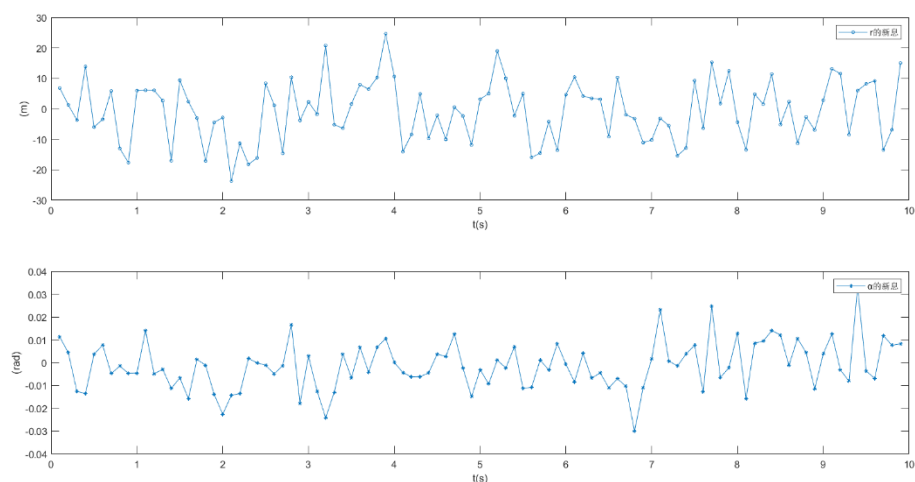


图 11 观测值逐次更新的 EKF 观测值新息

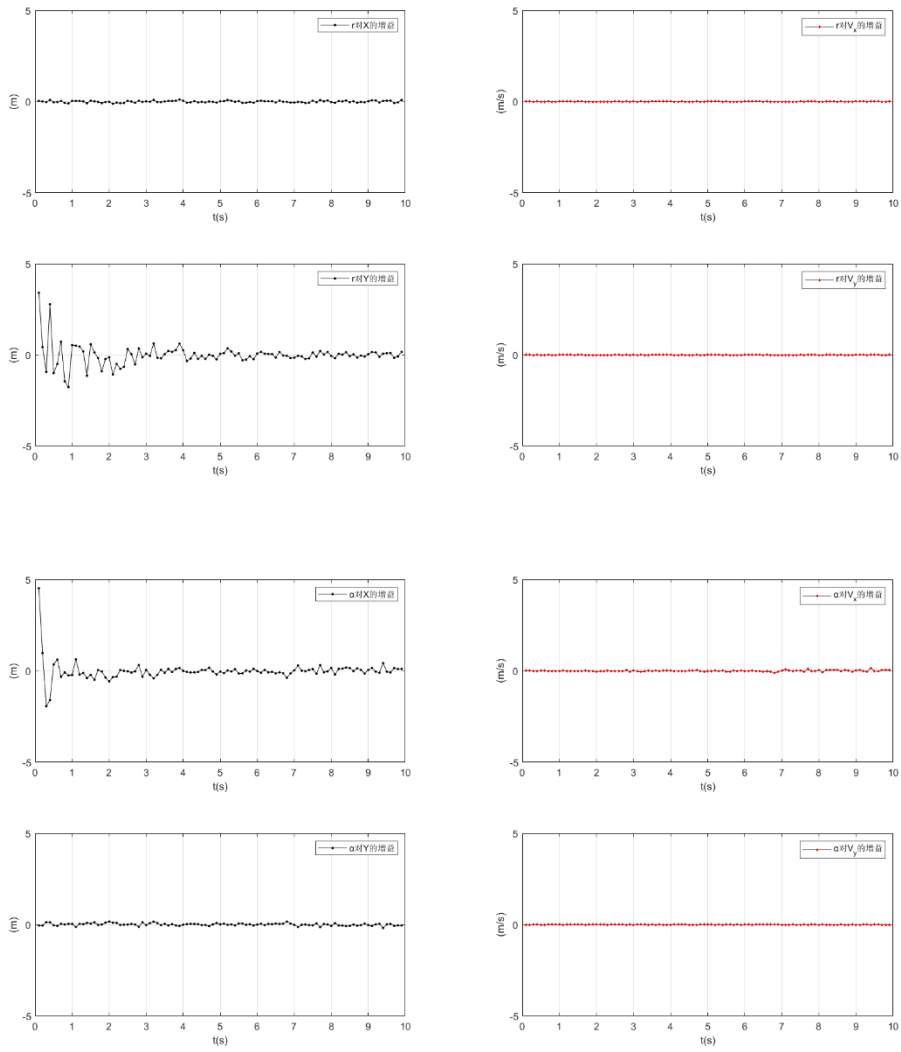


图 12 观测值逐次更新的 EKF 观测值对状态的增益

上图分别是观测值逐次更新的 EKF 观测值增益(图 11)和观测值逐次更新的 EKF 观测值对状态的增益(图 12)。观测值新息在 0 附近波动，由于观测值 α 的测量精度显著高于观测值 r ，因此 α 的新息波动幅度小于 r 的新息波动幅度。

通过绘制出两个观测值分别对状态的增益，不难发现观测值 r 对坐标 x 的增益小于对 y 的增益，而观测值 r 对坐标 y 的增益更大，体现出不同观测值与不同状态的相关性大小。

通过结果文件数据分析，扩展 Kalman 滤波和观测值逐次更新的 Kalman 滤波结果仅仅在小数部分存在差异，二者效果接近。

6 总结与反思

本次实验实现了利用 EKF 和观测值逐次更新的 EKF 对物体轨迹进行估计。相较于纸面的推导，利用编程解算的过程更能加深我们对 Kalman 滤波的理解。

利用扩展 Kalman 滤波和观测值逐次更新的扩展 Kalman 滤波进行解算时，结果上差异并不大。但是逐次更新的 EKF 在处理多维观测值且观测噪声互不相关的情况时具有更大的优势，因为逐次更新的 EKF 避免了矩阵求逆，数值稳定性更强，并且降低了协方差矩阵更新过程中正定性丢失的问题。但当处理观测维度低的情况时，传统 EKF 则具有更大的优势。

本次实验分别利用 C++ 和 Matlab 进行解算和可视化处理，提高了我们的编程能力和数据分析能力。

进行实验过程中也暴露了自身一些问题，例如对 EKF 实现过程不够熟悉，对解算结果的分析不够到位等不足。我也会在今后的学习中努力克服。