卡尔曼滤波作业2

一. 整理`EKF`的公式

1. 经典的`EKF`公式

`EKF`滤波公式可以分为时间更新和状态更新两个主要部分:

- 时间更新

$$\hat{x}_k^- = A\hat{x}_{k-1}^+ + Bu_{k-1}$$

$$P_k^- = AP_{k-1}^+A^T + Q$$

- 状态更新

$$K_k = P_k^- H^T(HP_k^- H^T + R)^{-1}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - H\hat{x}_k^-)$$

$$P_k^+ = (I - K_kH)P_k^-$$

这里需要说明一下，状态更新的第三个公式后验误差协方差矩阵的更新我看到过两种形式，上式是第一种，还有另外一种从最小二乘估计套用过来的更新后验误差协方差矩阵方法：

$$P_k^+ = (I - K_kH)P_k^-(I - K_kH)^T + K_1RK_1^T$$

我目前也不是很清楚这两种方式的差别，后续的实验中进行验证了，这两种方法应用在卡尔曼滤波上几乎没有区别

2. 状态空间模型

1. 状态量x
   本任务中使用的状态估计器估计的状态为世界坐标系$\{s\}$下的机器人运动状态，即机身位置p_b、机身速度v_b，这两个向量都是三维向量，所以状态向量x是一个6维向量:

$$x = \begin{bmatrix} p_b \\ v_b \end{bmatrix}$$

2. 连续状态空间方程
   状态向量x的导数为$\dot{x}$，所以$\dot{x}$为

$$\dot{x} = \begin{bmatrix} \dot{p_b} \\ \dot{v_b} \end{bmatrix} = \begin{bmatrix} v_b \\ R_{sb}a_b + g \end{bmatrix} = \begin{bmatrix} 0_{3*3} & I_{3*3} \\ \multicolumn{2}{c}{0_{3*6}} \end{bmatrix} x + \begin{bmatrix} 0_{3*3} \\ I_3 \end{bmatrix} [R_{sb}a_b + g]$$

3. 离散状态空间方程
   上面的这个是连续形式的状态空间模型，变成离散形式的状态空间方程之后为

$$x(k+1) = (A\Delta T + I)x(k) + B\Delta Tu(k)$$

$$x_{k+1} = \begin{bmatrix} I_{3*3} & \Delta t \cdot I_{3*3} \\ 0_{3*3} & I_{3*3} \end{bmatrix} x + \begin{bmatrix} 0_{3*3} \\ \Delta t \cdot I_3 \end{bmatrix} [R_{sb}a_{b_k} + g]$$

   这里我的方程和PPT里面的不太一致，主要就是离散状态下的u不太一致
   这里也有一点问题，代码中的B矩阵是这个形式的:

$$B = \begin{bmatrix} \frac{1}{2}\Delta t^2 \cdot I_3 \\ \Delta t \cdot I_3 \end{bmatrix}$$

   这里与我们从连续状态空间方程推导得到的离散状态空间方程并不一致，因此认为有些问题，后面再进行测试试一下~
   这里后面尝试了不太能测试出来，主要认为机器人一般还都是保持匀速的，因此影响不大，但是如果机器人是加速的跑，可能区别就打了，可以自己录制数据尝试一下

3. 观测方程

1. 输出向量y与观测方程

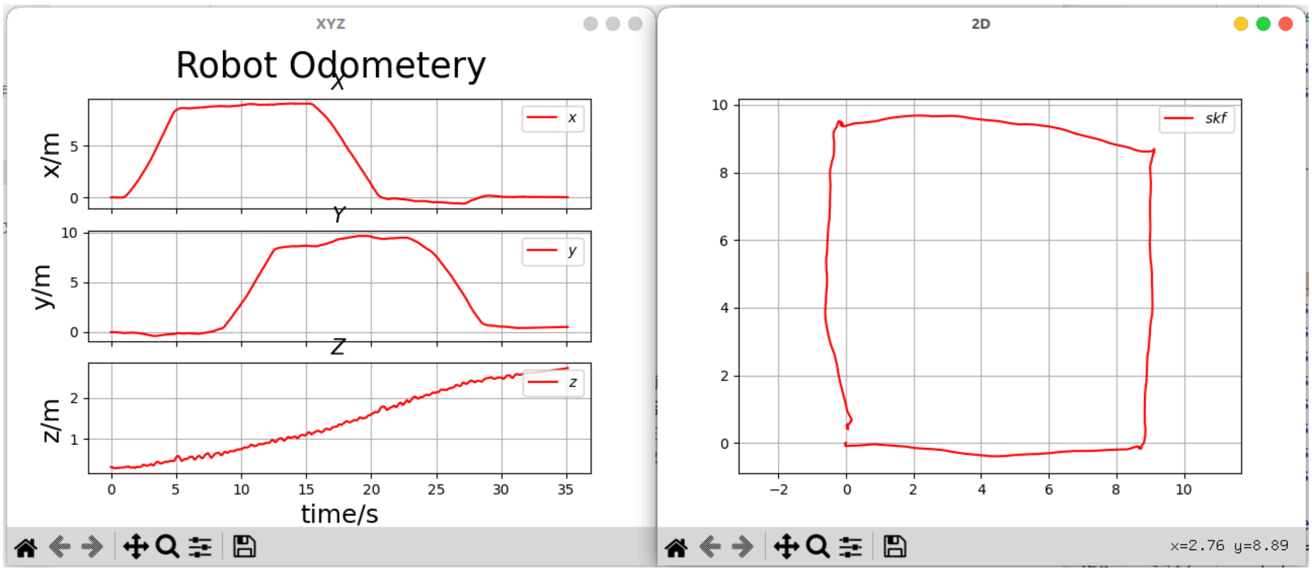
输出向量y中的物理量都是可以直接测量得到的，那么首先可以通过IMU得到世界坐标系$\{s\}$下的机身姿态R_{sb}，机身坐标系$\{b\}$下的加速度a_b和旋转角速度ω_b，其次，根据机器人编码器和正运动学，可以求得在机身坐标系$\{b\}$下的足端相对于机身的位置p_{bfB}，因为机身姿态R_{sb}已知，所以可以得到世界坐标系$\{s\}$下的足端相对于机身的位置p_{sfB}。

$$p_{sfB} = R_{sb} \cdot p_{bfB}$$

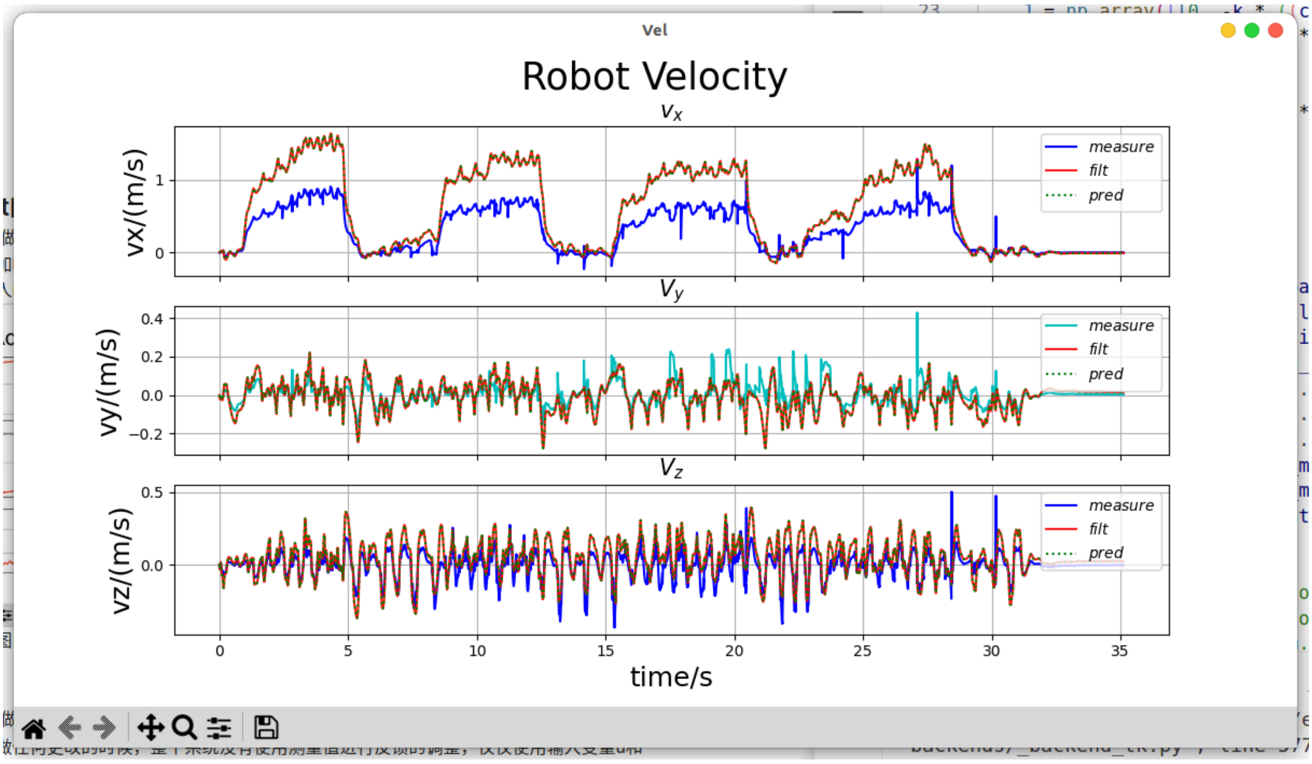
那么，在足端触地时，$-p_{sfB}$就是机身高度，可以通过每一个触地的足端分别解算出一个机身的高度，最后取平均值，当四条腿完全腾空时，记录上一个有足端触地的时间到现在的时间，然后使用自由落体公式得到当前时刻的机身高度作为机身高度的测量值。

$$y = \begin{bmatrix} p_{bz} \\ v_{bx} \\ v_{by} \\ v_{bz} \end{bmatrix} = Hx = [0_{4\times2} \quad I_{4\times4}]x = [0_{4\times2} \quad I_{4\times4}] \begin{bmatrix} p_{bx} \\ p_{by} \\ p_{bz} \\ v_{bx} \\ v_{by} \\ v_{bz} \end{bmatrix}$$

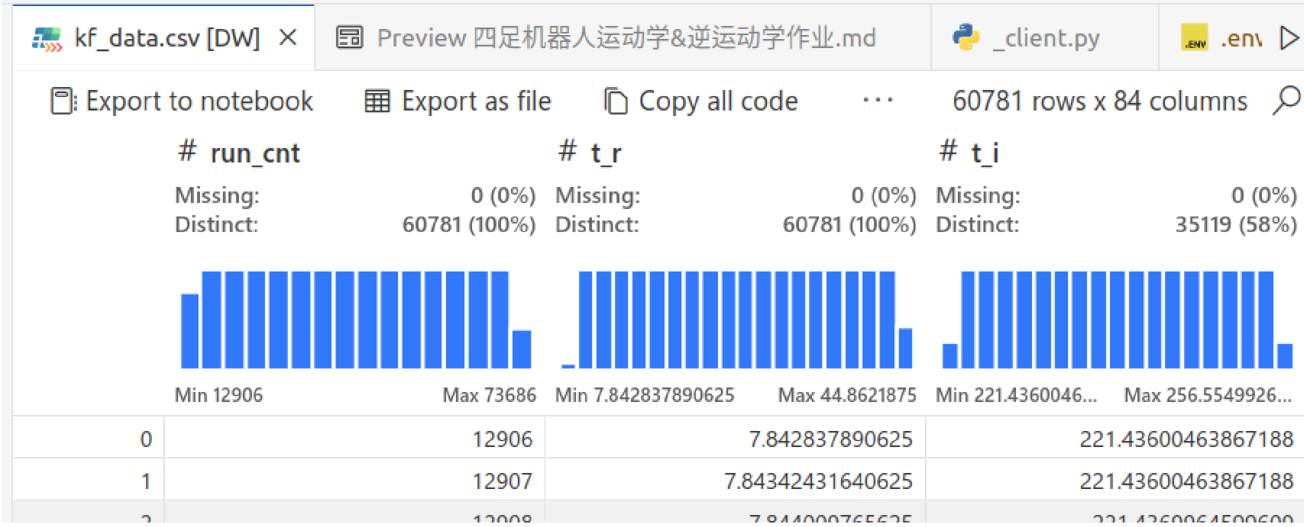
二. Project的代码效果

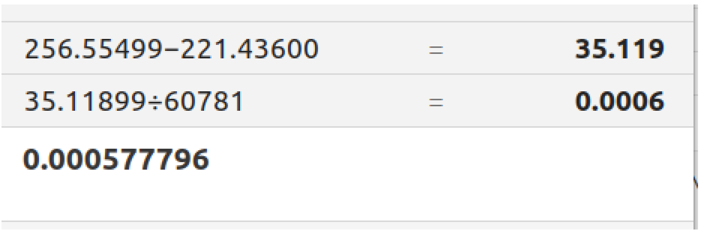
1. 源代码(未做改变)，使用初始值+无人为增加噪声+模型进行预测
   在没有人为增加噪声的时候，整个系统的传感器精度其实还相对不错，可以大致地使用系统模型推断出足式机器人的移动情况如下图所示：
   

   但是发现这个图，系统对于状态的预测出现了很大的偏差，主要是前进方向，因为左右横移和高度变化其实本来也没有多少，主要还是在0附近

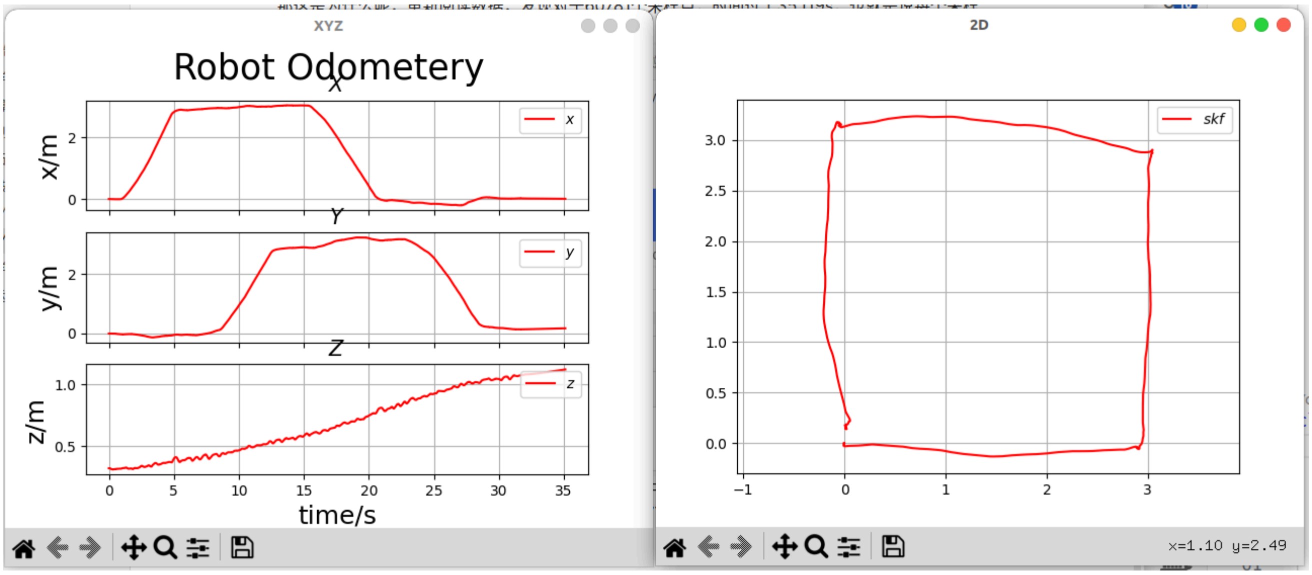


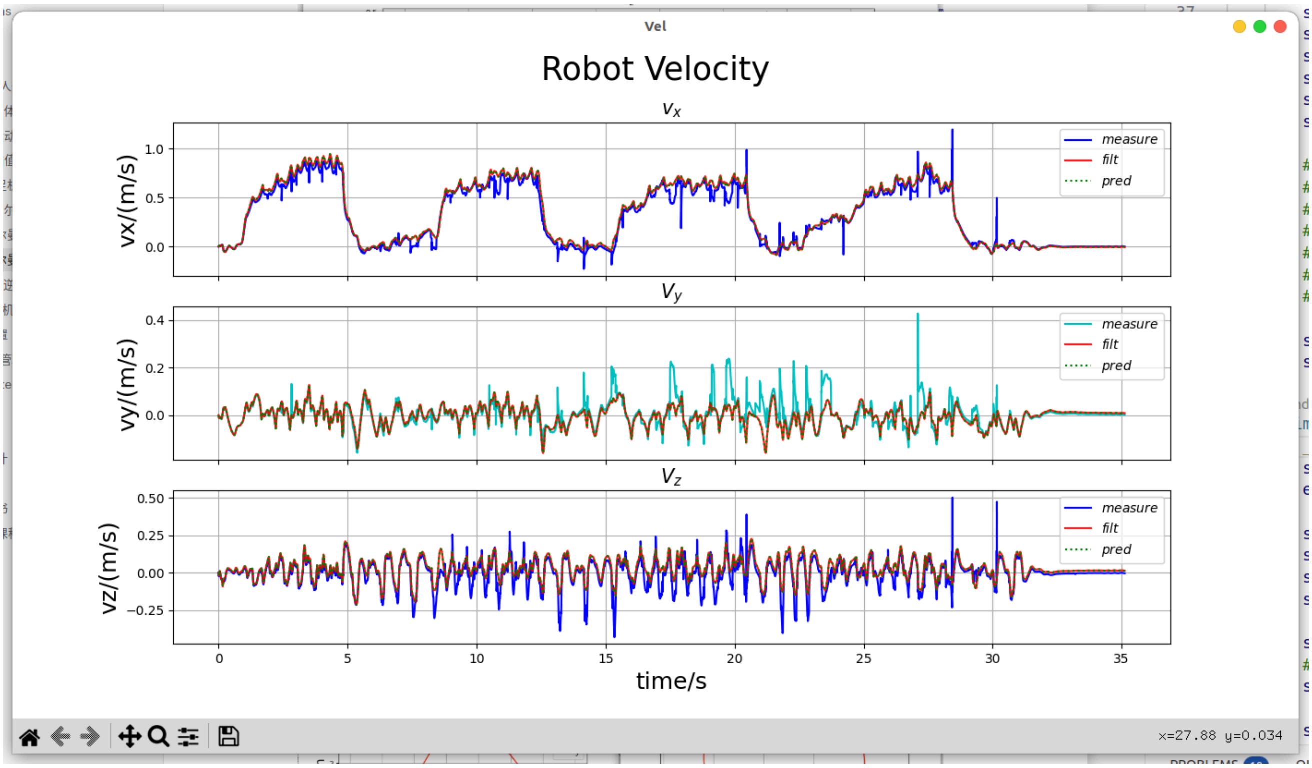
那这是为什么呢，重新阅读数据，发现对于60781个采样点，时间过了35.119s，也就是说每个采样点的时间间隔其实均值为0.000577796

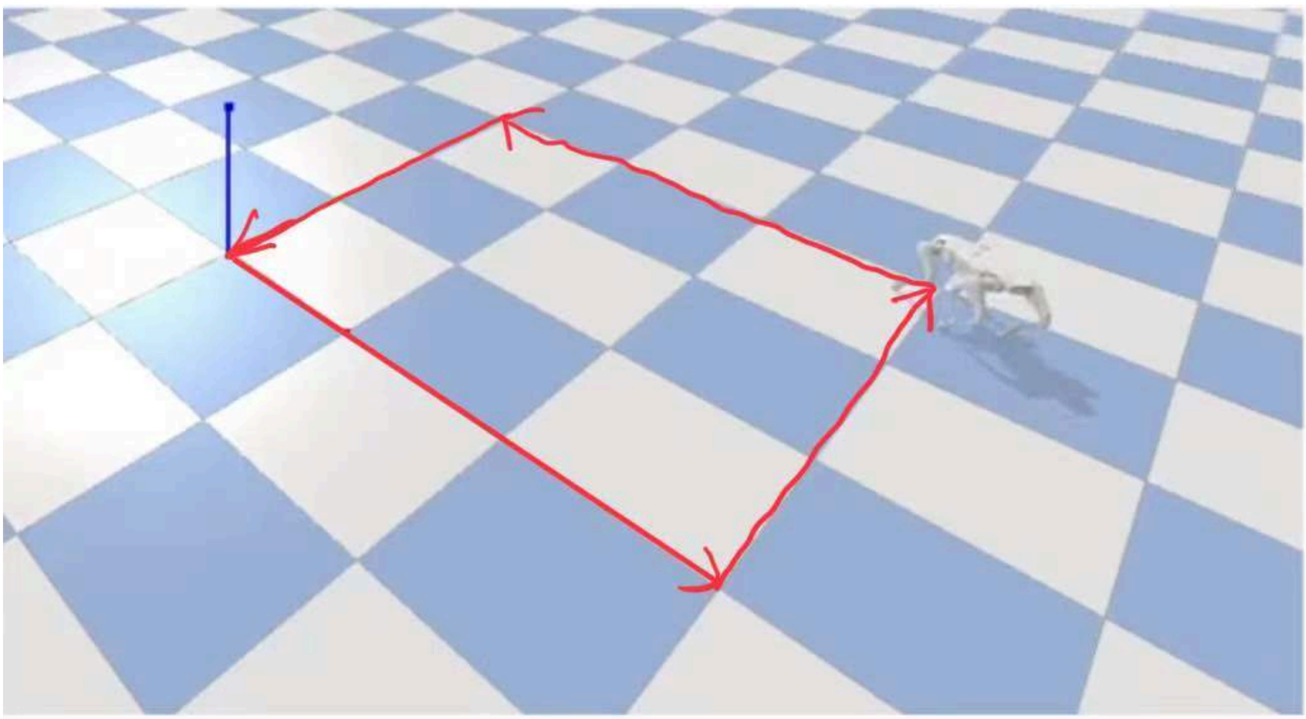




如果这个数据不正确的话，就会显著地影响离散控制中机器人的δt，会显著地影响参数的幅值，因此尝试将卡尔曼滤波器的时间间隔δt更改为0.000577796，重新运行，得到的数据如下图所示，从下图中可以发现，主要表示行进方向的x方向的速度目前预测的非常好，y和z方向也只是有一些高频的噪声没有预测，不影响大局，现在的主要区别是机器人的行进变成了一个$3m \times 3m$见方的正方形，看了一些课程的Project的介绍，符合其中狗的运动距离。


XYZ
Robot Odometery
X
x/m
x
Y
y/m
y
Z
z/m
z
time/s
2D
skf
x=1.10 y=2.49

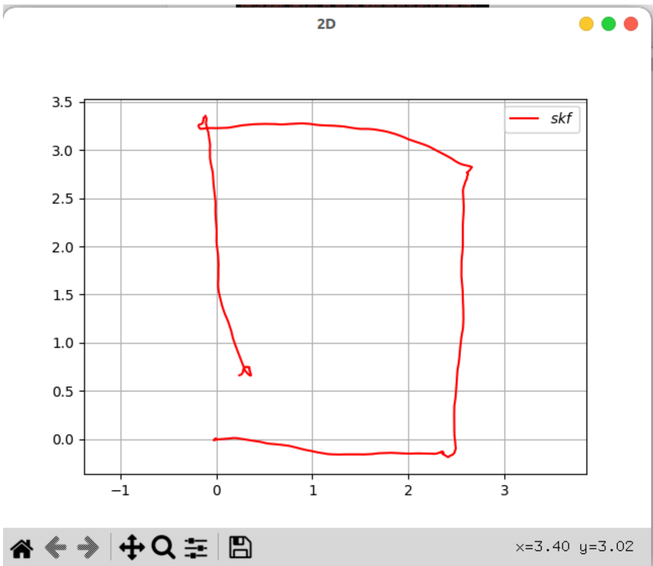

Vel
Robot Velocity
v_x
vx/(m/s)
measure
filt
pred
V_y
vy/(m/s)
measure
filt
pred
V_z
vz/(m/s)
measure
filt
pred
time/s
x=27.88 y=0.034

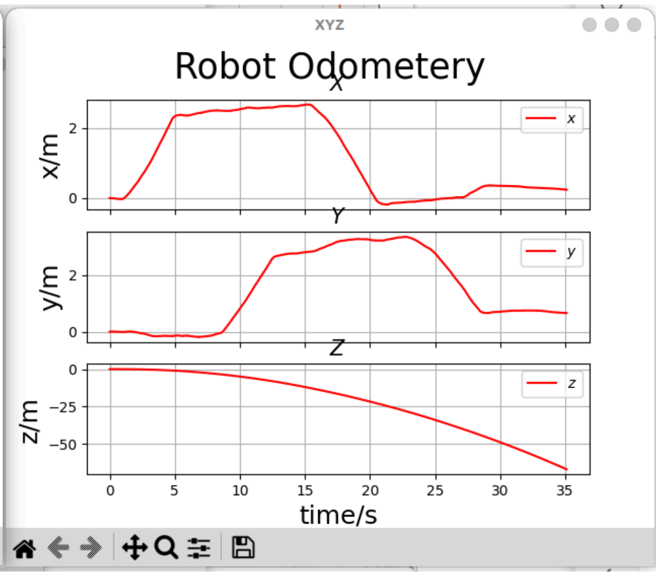


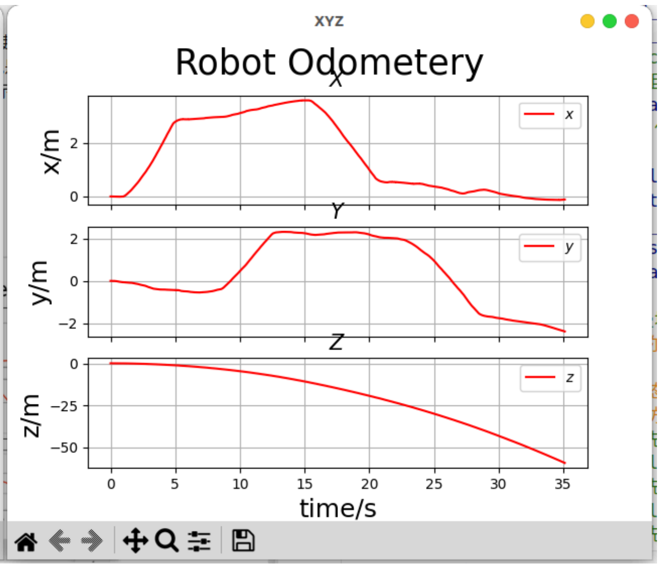
2. 使用初始值+噪声+模型进行预测

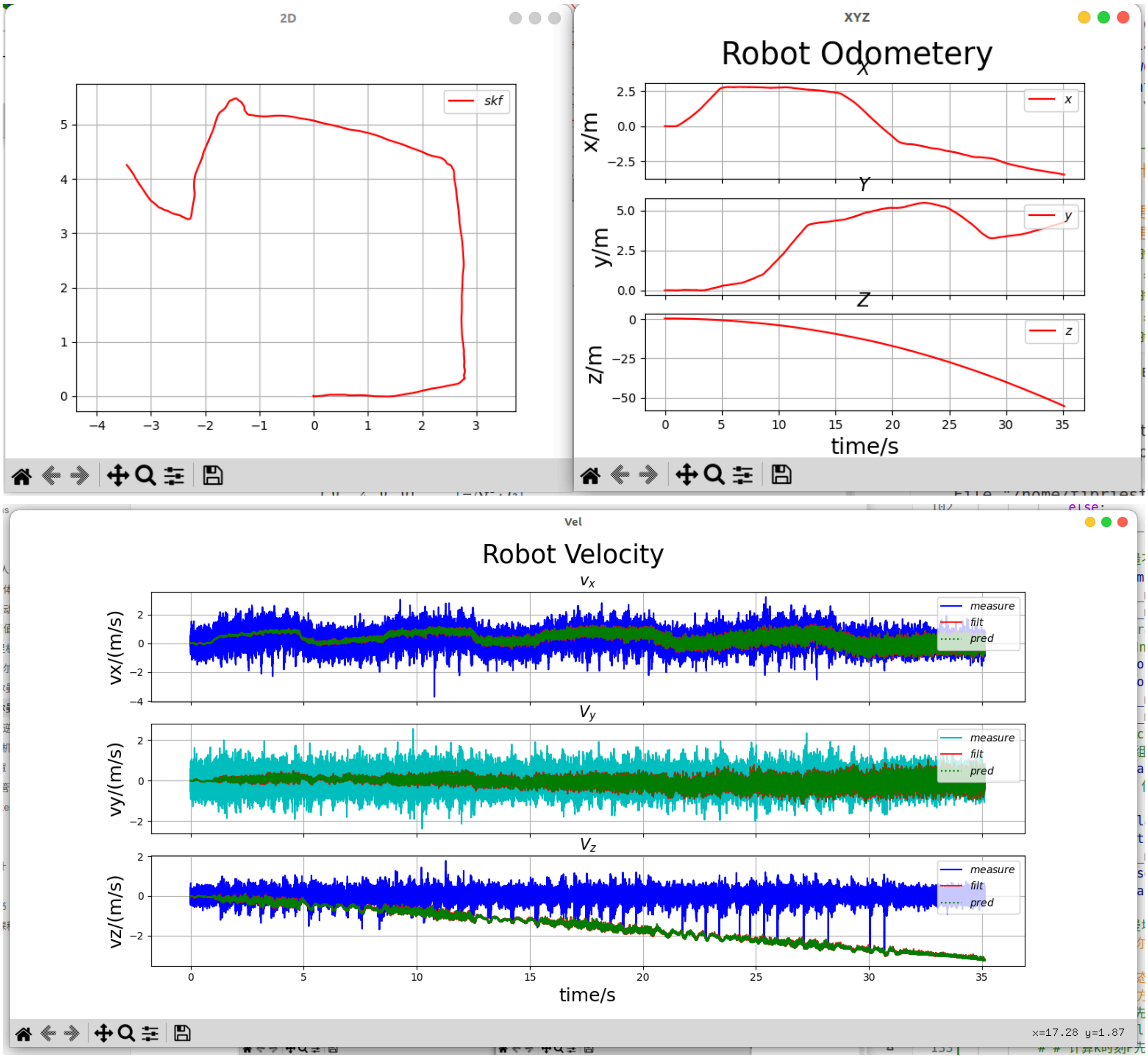
在代码完全不做任何更改的时候，整个系统没有使用测量值进行反馈的调整，仅仅使用输入变量u和之前的状态x来对系统进行控制，因此由于系统加入的噪声的存在，整个形式只能说是大体形式上相同，但是最后的结尾的位置还是会有很多不同可能的误差，系统增加的噪声如下所示：

$$
\begin{aligned}
addNoise(rpy) &= Normal(0, 0.1) \\
addNoise(acc) &= Normal(0, 1.0) \\
addNoise(w) &= Normal(0, 0.8) \\
addNoise(joint\_pos) &= Normal(0, 0.03) \\
addNoise(joint\_vel) &= Normal(0, 1.6) \\
addNoise(joint\_tau) &= Normal(0, 1.0)
\end{aligned}
$$





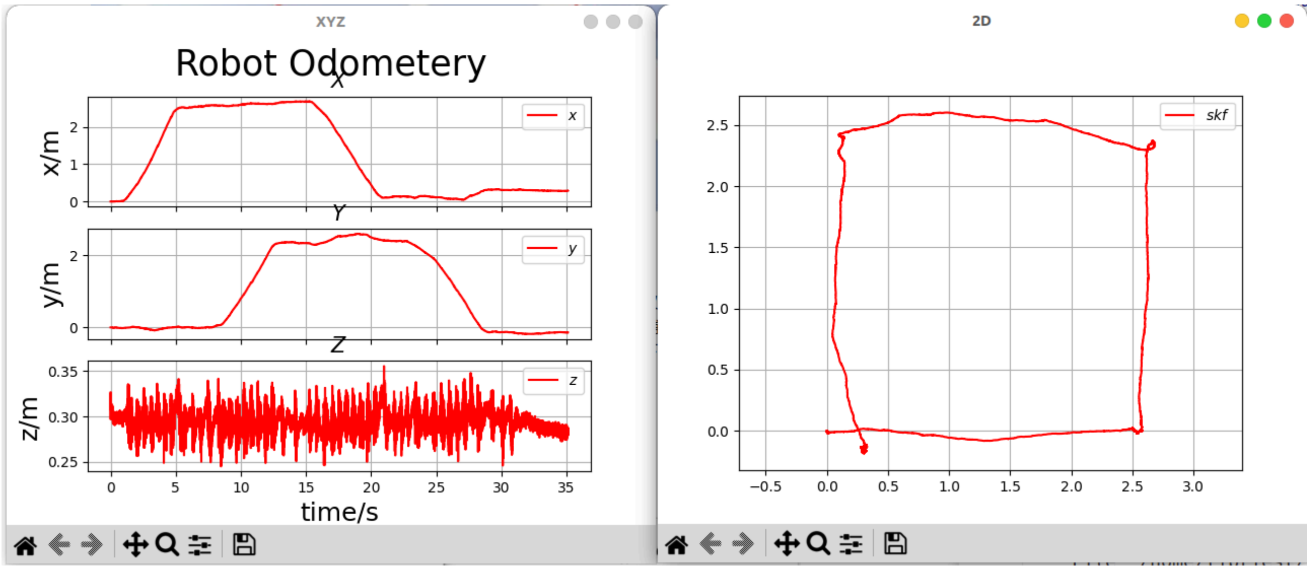


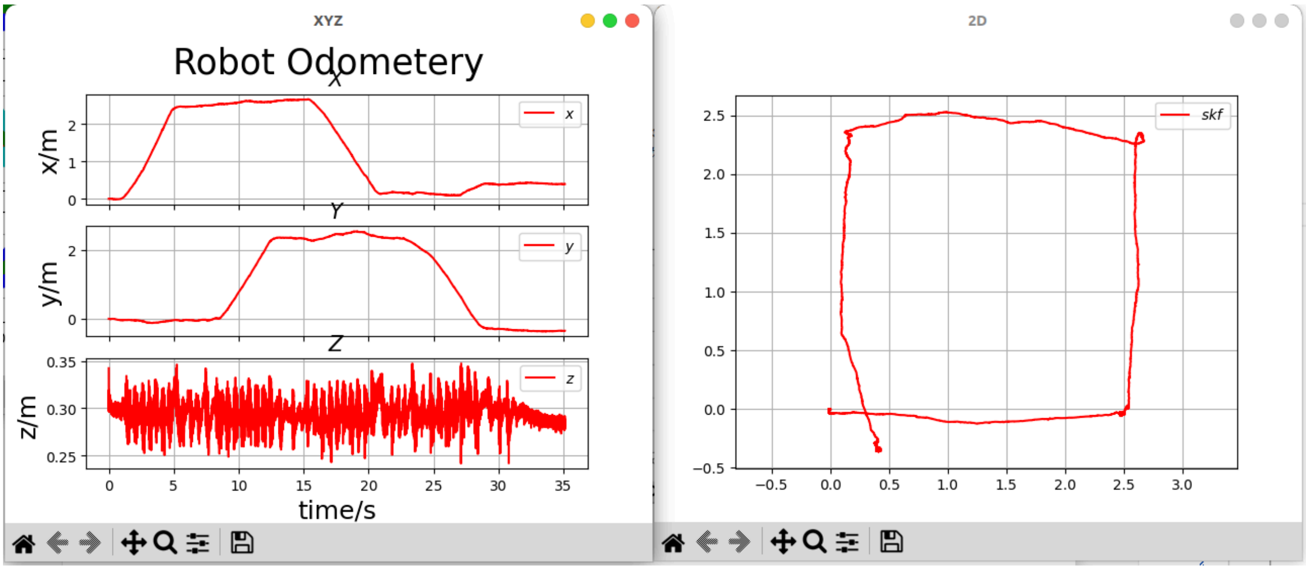


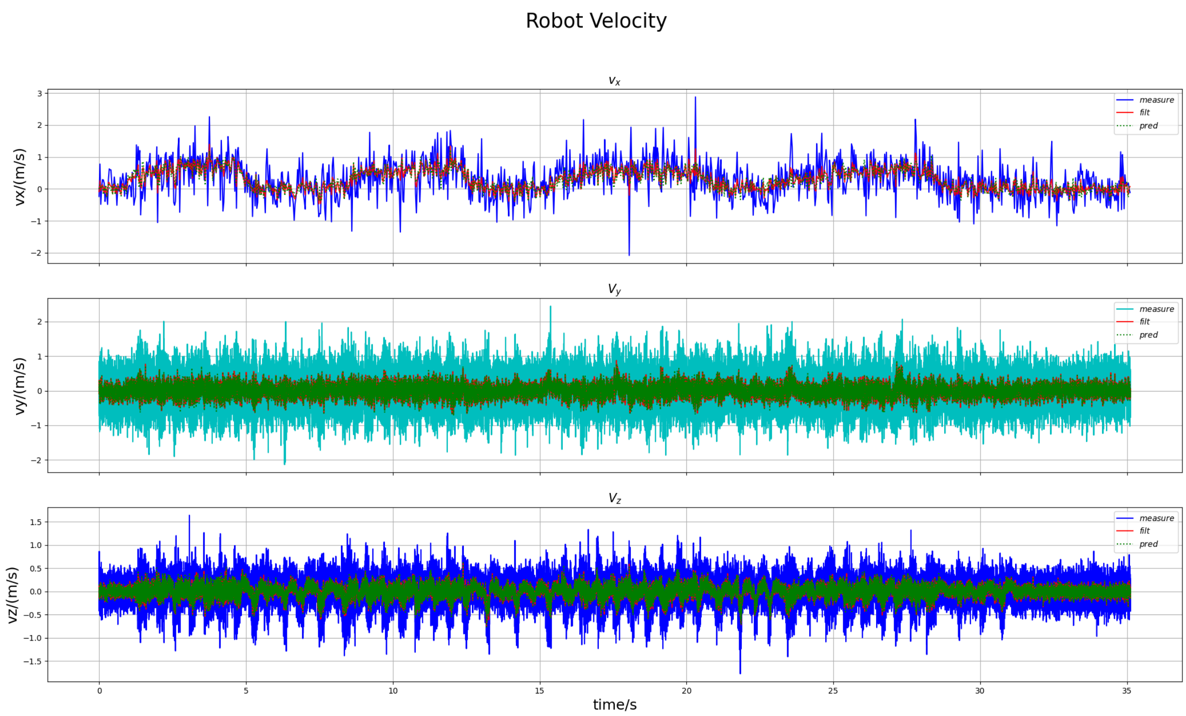
3. 使用初始值+噪声+模型+卡尔曼滤波进行预测
   使用了卡尔曼滤波之后的情况，使用相同的参数进行多组实验，观察最后的x，y和z的位置
   初始参数：

$$P_0=\begin{bmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\end{bmatrix},Q=\begin{bmatrix}0.1&0&0&0&0&0\\0&0.1&0&0&0&0\\0&0&0.1&0&0&0\\0&0&0&0.1&0&0\\0&0&0&0&0.1&0\\0&0&0&0&0&0.1\end{bmatrix}$$

$$R=\begin{bmatrix}0.5&0&0&0\\0&2&0&0\\0&0&2&0\\0&0&0&2\end{bmatrix},B=\begin{bmatrix}0_{3\times3}\\\Delta t\cdot I_3\end{bmatrix}$$





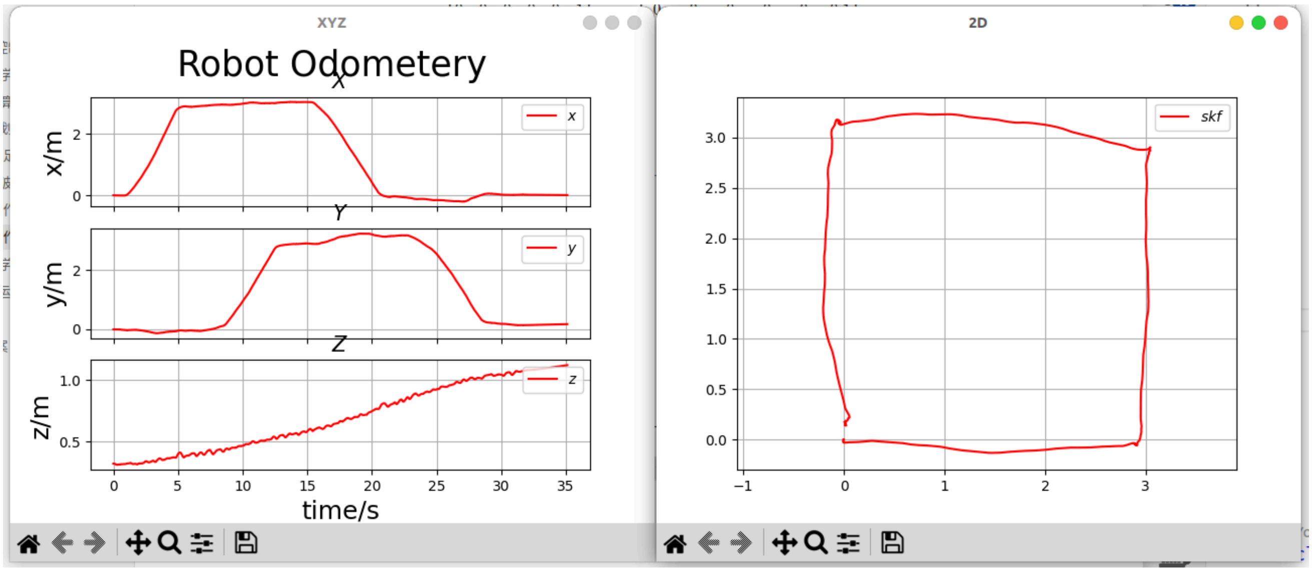


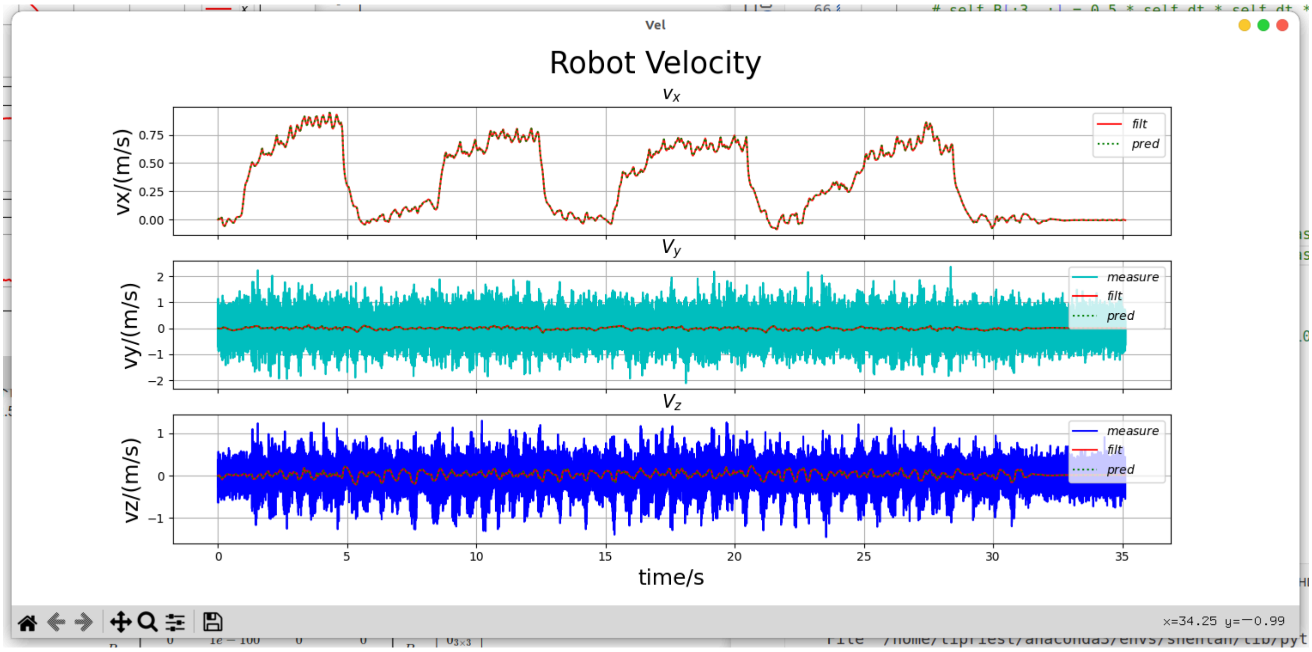
4. 卡尔曼滤波参数调整

在3中发现系统在使用卡尔曼滤波之后虽然还原了大致的运动趋势，但是整体预测之后走的距离只有2.6m左右，是卡尔曼滤波器写的有问题吗？还是由于什么原因导致的最后走的距离不够呢？这里采用比较极端的验证方法，分别将R->inf, Q->0和R->0, Q->inf进行实验，也就是只取预测值和只取测量值进行实验，在实验前将输入的噪声取0，否则对预测造成的不确定性过大难以分辨，将输入的噪声取0，在极限环境下应该和1中的状态相似。

$$P_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, Q = \begin{bmatrix} 1e-100 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1e-100 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1e-100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1e-100 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1e-100 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1e-100 \end{bmatrix}$$

$$R = \begin{bmatrix} 1e100 & 0 & 0 & 0 \\ 0 & 1e100 & 0 & 0 \\ 0 & 0 & 1e100 & 0 \\ 0 & 0 & 0 & 1e100 \end{bmatrix}, B = \begin{bmatrix} 0_{3\times 3} \\ \Delta t \cdot I_3 \end{bmatrix}$$

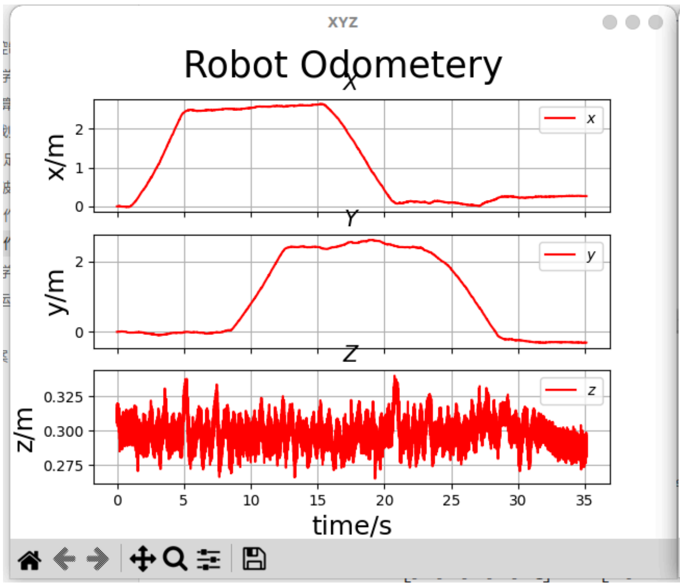


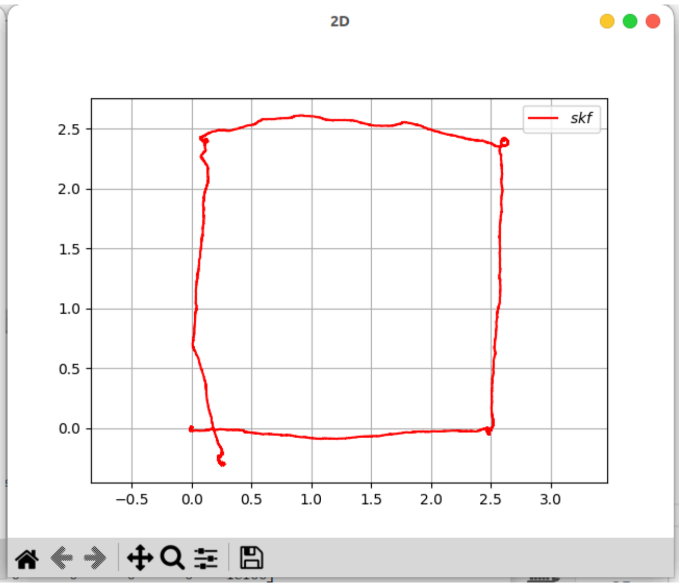


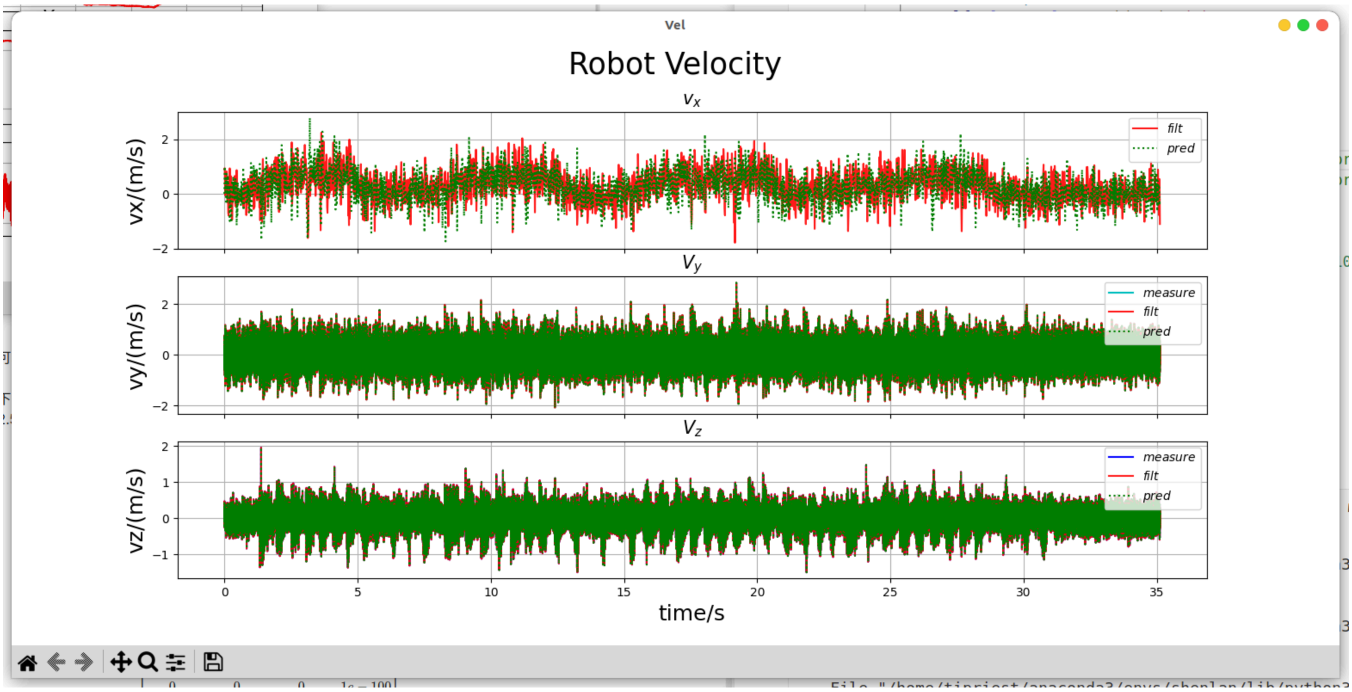
可以看到，在这种极端情况下成功复现了1中的效果，走的距离在3m多一点，形状近乎完全相似。
下面验证一下只使用测量值的情况，参数如下，可以看到只使用测量值的情况下行走的距离大致在2.5m左右。

$$P_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, Q = \begin{bmatrix} 1e100 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1e100 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1e100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1e100 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1e100 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1e100 \end{bmatrix}$$

$$R = \begin{bmatrix} 1e-100 & 0 & 0 & 0 \\ 0 & 1e-100 & 0 & 0 \\ 0 & 0 & 1e-100 & 0 \\ 0 & 0 & 0 & 1e-100 \end{bmatrix}, B = \begin{bmatrix} 0_{3\times 3} \\ \Delta t \cdot I_3 \end{bmatrix}$$







5. 总结

因此，3中的结论无误，验证了卡尔曼滤波公式确实融合了模型的预测值和测量值，输出了合理的轨迹，其实从2，3，4的v_y,v_z中能够非常明显地看出来，测量值因为方差的存在变成了很宽的一个区域，在使用预测值的时候，估计值几乎为0，在使用测量值的时候，与测量值基本重合。

三. (附加题)扩展卡尔曼滤波

to be continued...先交作业