

EKF 小结

EKF 滤波在工程实现上采用较多，近期使用较多，编写一个小结文档用于梳理知识和后期使用快速上手。主要是总结现有 EKF 使用案例。

1 案例一（二阶 KF）

二阶 EKF 主要是用于划线车喷口航向角的检测，喷口上自带 IMU，可获取喷口的角速度，同时丝杆移动可以换算成喷口角度，划线车移动时会改变航向角，两个角度叠加可获得喷口相对正北的航向角。根据实际情况决定使用二阶 EKF 来获取喷口的角度和角速度，角速度目前可以直接使用 IMU 均值滤波获取，角度需要使用 KF 滤波获取，短时信任角速度积分获得角速度，长期则信任丝杆换算的角度。

本次状态有两个，分别是角度(yaw)和角度变化量 bias errors (daz_b)，第二个状态主要是用于修正角速度的偏差。

1.1 状态方程

状态方程如下：

$$\text{stateVector} = \begin{bmatrix} \text{yaw} \\ \text{daz_b} \end{bmatrix}$$

$$\text{stateVectorNew} = \begin{bmatrix} \text{daz} - \text{daz_b} + \text{yaw} \\ \text{daz_b} \end{bmatrix}$$

其中，daz 代表角度变化，来自于角速度积分，可以当成噪声处理。

使用常用公式对上述方程简化：

$$x_{k+1} = F \cdot x_k + G \cdot n_k$$

其中 $F = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}$ ， $G = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ ， $n_k = \Delta\theta_{nz} = \text{daz}$ ，航向角度的变化

$$Q = G \cdot \text{dazVar} \cdot G^T = \begin{bmatrix} \text{dazVar} & 0 \\ 0 & 0 \end{bmatrix}$$

$\text{dazVar} = \text{daz}^2$ 为角度变化噪声方差

1.2 观测方程

$$z=H \cdot x+R \Rightarrow \text{YawMeas} = \text{yaw}$$

$H=H_ACC=[1, 0]$ ， R 是当前时刻 Yaw 观测的不确定度，即 Yaw 的协方差。

1.3 预测

1.3.1 状态预测

状态预测相对简单，根据状态方程更新状态即可：

$$\text{stateVectorNew} = \begin{bmatrix} \text{daz} - \text{daz_b} + \text{yaw} \\ \text{daz_b} \end{bmatrix}$$

1.3.2 协方差预测

协方差预测较为复杂，目前只列出公式：

$$P = F \cdot P \cdot F^T + Q + N_{\text{process}}$$

N_{process} 为除 IMU 噪声以外的滤波器状态过程误差，称过程噪声协方差

$$F = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}, \quad Q = \begin{bmatrix} \text{dazVar} & 0 \\ 0 & 0 \end{bmatrix}, \quad N_{\text{process}} = [0, dAngBiasSigma].^2$$

1.4 观测

套用以下公式即可：

$$S = H \cdot P_{k+1|k} \cdot H^T + R$$

$$K = P_{k+1|k} \cdot H^T \cdot S^{-1}$$

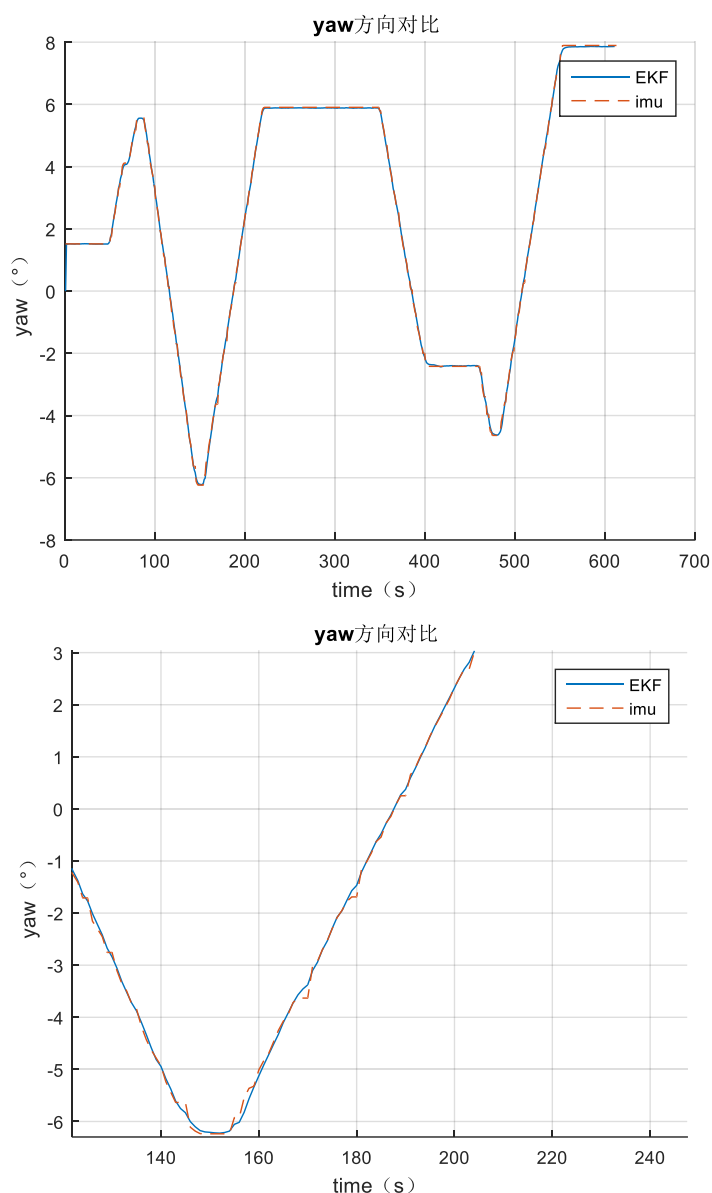
$$x_{k+1|k+1} = x_{k+1|k} + K \cdot (z - H \cdot x_{k+1|k})$$

$$P_{k+1|k+1} = [I - K \cdot H] \cdot P_{k+1|k}$$

其中， $H=[1, 0]$ ， $R = \text{YawNoise}$ ， $z=\text{YawMeas}$

1.5 仿真结果

下图是仿真结果，图 1 可以看出 EKF 输出整体效果非常好，图 2 则是图 1 的放大部分观测角度存在不连续的过程，经过滤波之后较为平滑，满足短期信任角速度积分获得角度，长期信任丝杆转换的角度。



2 案例二 （五阶 EKF）

五阶 EKF 主要是用于工具车导航，传感器有 IMU，ODO 和激光雷达（二维码），用于输出工具车的航向角，水平位置和速度。本次激光雷达输出的为水平位置和航向角度，但是输出角度连续性较差，出现跳点，如果直接用于控制会影响控制效果，所以需要激光雷达输出数据进行滤波。滤波之前需要对角速度和速度进行标定，确保短时间积分数据较为准确，长时间则相信激光雷达数据。

2.1 状态方程

状态向量如下：

$$\text{stateVector} = \begin{bmatrix} px \\ py \\ yaw \\ v \\ w \end{bmatrix}$$

其中， px 代表 x 方向水平位置；

py 代表 y 方向水平位置；

yaw 代表工具车的航向角；

v 代表工具车的前向速度；

w 代表工具车的航向角速度；

状态更新方程如下：

$$\text{stateVectorNew} = \begin{bmatrix} px + dt * pxn + dt * v * \cos(yaw) \\ py + dt * pyn + dt * v * \sin(yaw) \\ (wn * dt^2) / 2 + w * dt + yaw \\ v + dt * vn \\ w + dt * wn \end{bmatrix}$$

其中， pxn 代表 x 方向水平位置的速度噪声；

pyn 代表 y 方向水平位置的速度噪声；

wn 代表航向角加速度噪声；

vn 代表前向水平加速度噪声；

dt 代表两次更新间隔时间；

使用常用公式对上述方程简化：

$$x_{k+1} = F \cdot x_k + G \cdot n_k$$

$$\text{其中 } F = \begin{bmatrix} 1, 0, -dt*v*\sin(yaw), dt*\cos(yaw), 0 \\ 0, 1, dt*v*\cos(yaw), dt*\sin(yaw), 0 \\ 0, 0, 1, 0, dt \\ 0, 0, 0, 1, 0 \\ 0, 0, 0, 0, 1 \end{bmatrix}, \quad G = \begin{bmatrix} dt, 0, 0, 0 \\ 0, dt, 0, 0 \\ 0, 0, 0, dt^2/2 \\ 0, 0, dt, 0 \\ 0, 0, 0, dt \end{bmatrix}, \quad n_k = \begin{bmatrix} pxn \\ pyn \\ vn \\ wn \end{bmatrix}$$

$$Q = G \cdot \text{distMatrix} \cdot G^T = \begin{bmatrix} dt^2 * pxVar, 0, 0, 0, 0 \\ 0, dt^2 * pyVar, 0, 0, 0 \\ 0, 0, (dt^4 * wVar) / 4, 0, (dt^3 * wVar) / 2 \\ 0, 0, 0, dt^2 * vVar, 0 \\ 0, 0, (dt^3 * wVar) / 2, 0, dt^2 * wVar \end{bmatrix}$$

$$\text{distMatrix} = \begin{bmatrix} pxVar, & 0, & 0, & 0 \\ 0, & pyVar, & 0, & 0 \\ 0, & 0, & vVar, & 0 \\ 0, & 0, & 0, & wVar \end{bmatrix} \text{ 为 } n_k \text{ 噪声方差}$$

其中， $pxVar$ 代表 x 方向水平位置的速度噪声的方差；

$pyVar$ 代表 y 方向水平位置的速度噪声的方差；

$wVar$ 代表航向角加速度噪声的方差；

$vVar$ 代表前向水平加速度噪声的方差；

上述参数调大，则噪声很大，预测信任度降低，观测信任度提高。

2.2 观测方程

2.2.1 Odo 观测

$$z = H \cdot x + R$$

$$z = \begin{bmatrix} \text{Vodom} \\ \text{Wodom} \end{bmatrix} = \begin{bmatrix} v \\ w \end{bmatrix}$$

$$H = H_{\text{ODO}} = \begin{bmatrix} 0, 0, 0, 1, 0 \\ 0, 0, 0, 0, 1 \end{bmatrix}$$

R 是当前时刻观测的不确定度，即 odo 的协方差。

2.2.1 imu 观测

$$z = H \cdot x + R$$

$$z = [\text{Wimu}] = [w]$$

$$H = H_{\text{IMU}} = [0, 0, 0, 0, 1]$$

R 是当前时刻观测的不确定度，即 IMU 的协方差。

2.2.1 lidar 观测

$$z = H \cdot x + R$$

$$z = \begin{bmatrix} \text{lidar_x} \\ \text{lidar_y} \\ \text{lidar_yaw} \end{bmatrix} = \begin{bmatrix} \text{px} \\ \text{py} \\ \text{yaw} \end{bmatrix}$$

$$H=H_Lidar = \begin{bmatrix} 1, 0, 0, 0, 0 \\ 0, 1, 0, 0, 0 \\ 0, 0, 1, 0, 0 \end{bmatrix}$$

R 是当前时刻观测的不确定度，即 lidar 的协方差。

2.3 预测

2.3.1 状态预测

状态预测相对简单，根据状态方程更新状态即可：

$$\text{stateVectorNew} = \begin{bmatrix} \text{px} + dt * \text{pxn} + dt * v * \cos(\text{yaw}) \\ \text{py} + dt * \text{pyn} + dt * v * \sin(\text{yaw}) \\ (\text{wn} * dt^2) / 2 + w * dt + \text{yaw} \\ v + dt * \text{vn} \\ w + dt * \text{wn} \end{bmatrix}$$

2.3.2 协方差预测

协方差预测较为复杂，目前只列出公式：

$$P = F \cdot P \cdot F^T + Q + N_{\text{process}}$$

N_{process} 为除 IMU 噪声以外的滤波器状态过程误差，称过程噪声协方差

$$F = \begin{bmatrix} 1, 0, -dt*v*\sin(\text{yaw}), dt*\cos(\text{yaw}), 0 \\ 0, 1, dt*v*\cos(\text{yaw}), dt*\sin(\text{yaw}), 0 \\ 0, 0, 1, 0, dt \\ 0, 0, 0, 1, 0 \\ 0, 0, 0, 0, 1 \end{bmatrix}, \quad Q = \begin{bmatrix} dt^2 * \text{pxVar}, 0, 0, 0, 0 \\ 0, dt^2 * \text{pyVar}, 0, 0, 0 \\ 0, 0, (dt^4 * wVar) / 4, 0, (dt^3 * wVar) / 2 \\ 0, 0, 0, dt^2 * vVar, 0 \\ 0, 0, (dt^3 * wVar) / 2, 0, dt^2 * wVar \end{bmatrix},$$

$$N_{\text{process}} = [0, 0, 0, \text{dVelBiasSigma}, \text{dAngBiasSigma}].^2$$

2.4 观测

套用以下公式即可：

$$S = H \cdot P_{k+1|k} \cdot H^T + R$$

$$K = P_{k+1|k} \cdot H^T \cdot S^{-1}$$

$$x_{k+1|k+1} = x_{k+1|k} + K \cdot (z - H \cdot x_{k+1|k})$$

$$P_{k+1|k+1} = [I - K \cdot H] \cdot P_{k+1|k}$$

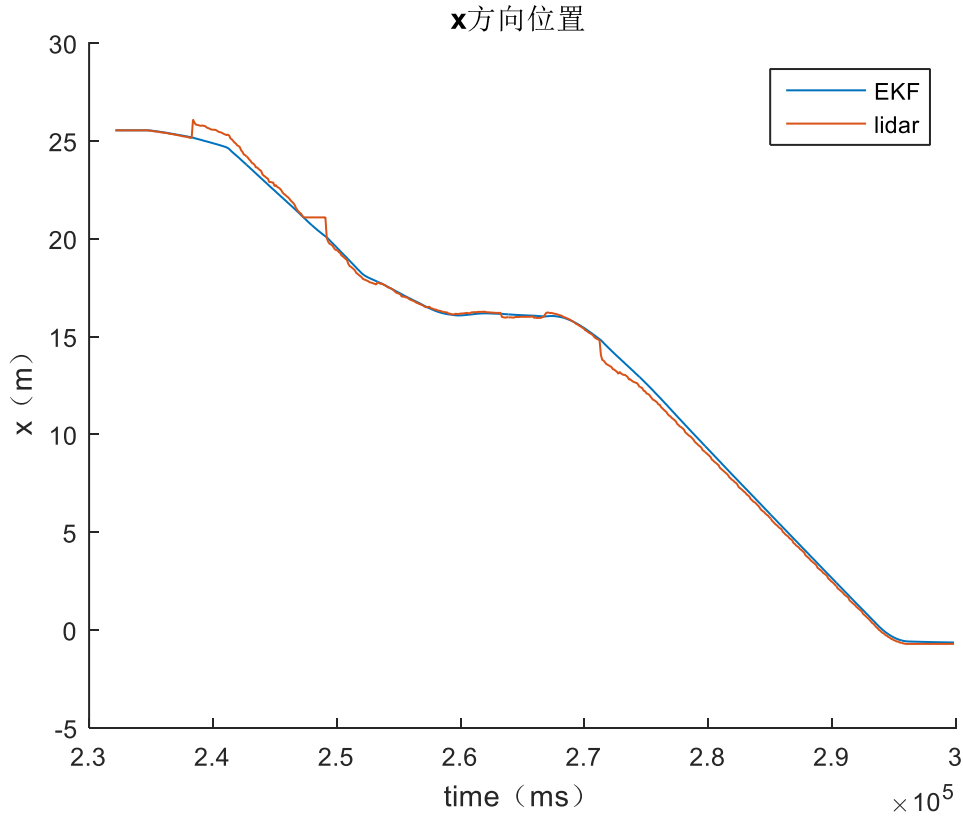
其中, $H_{\text{ODO}} = \begin{bmatrix} 0, 0, 0, 1, 0 \\ 0, 0, 0, 0, 1 \end{bmatrix}$, $R = \begin{bmatrix} \text{OdomVNoise} \\ \text{OdomWNoise} \end{bmatrix}$, $z = \begin{bmatrix} \text{odom_v} \\ \text{odom_w} \end{bmatrix}$

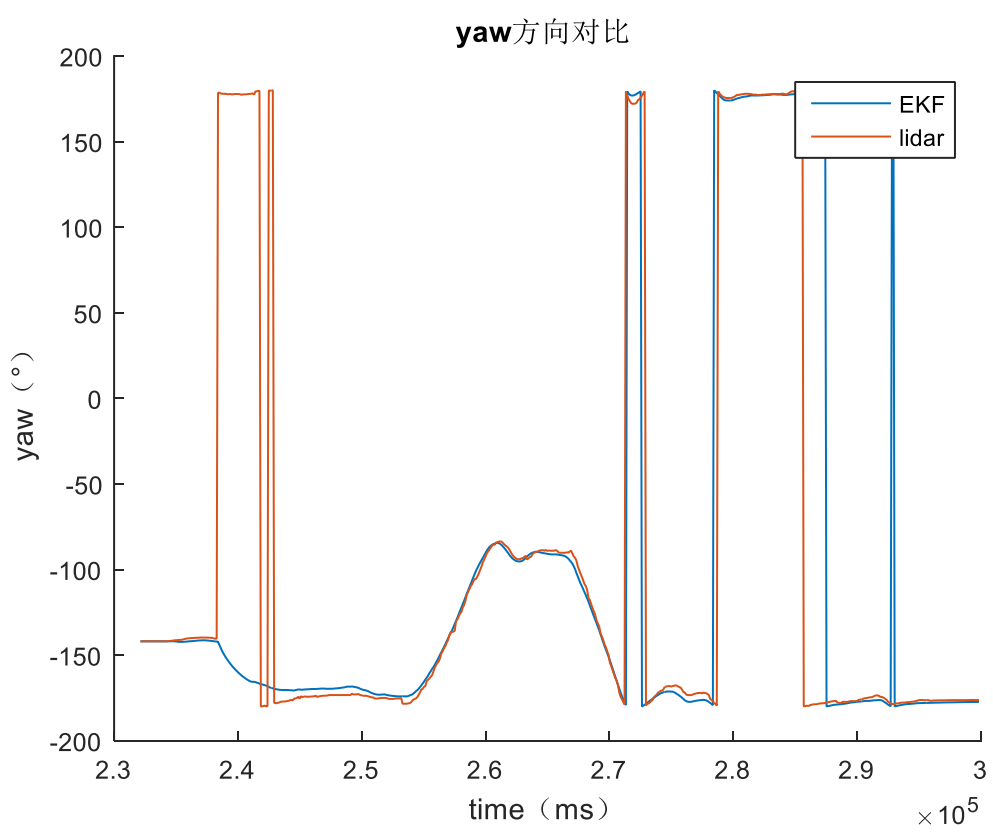
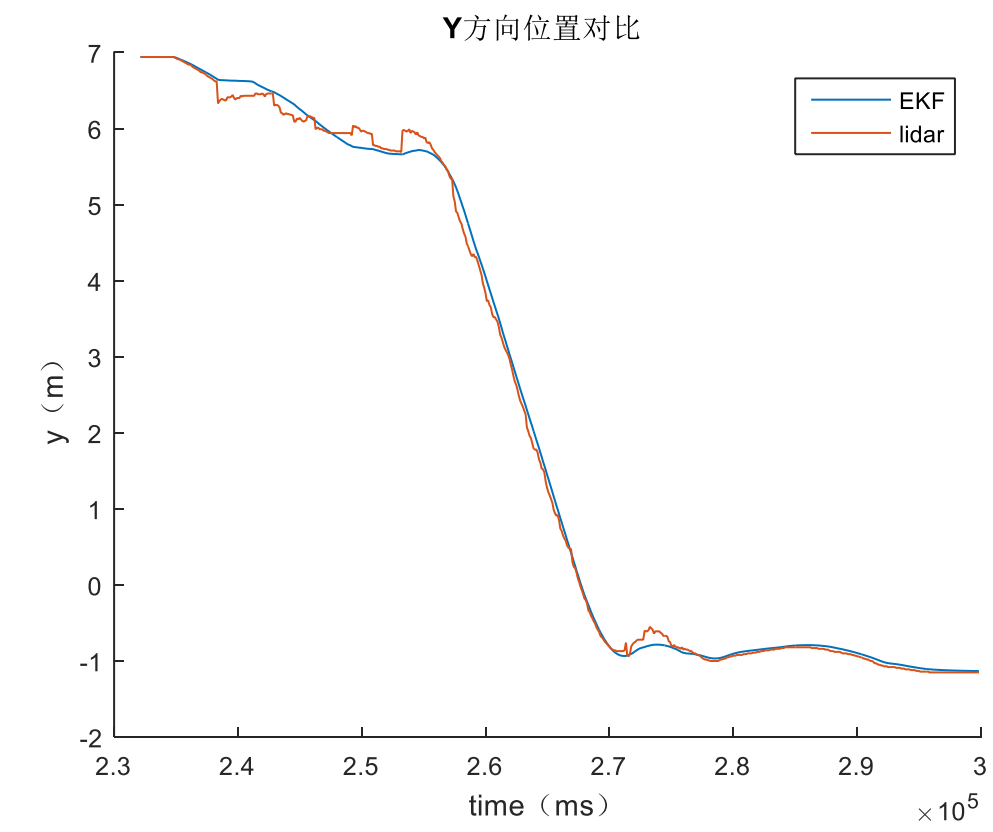
$$H_{\text{IMU}} = [0, 0, 0, 0, 1], R = [\text{gyroYawNoise}], z = [\text{w_yaw}]$$

$$H_{\text{Lidar}} = \begin{bmatrix} 1, 0, 0, 0, 0 \\ 0, 1, 0, 0, 0 \\ 0, 0, 1, 0, 0 \end{bmatrix}, R = \begin{bmatrix} \text{LidarXNoise} \\ \text{LidarYNoise} \\ \text{LidarYawNoise} \end{bmatrix}, z = \begin{bmatrix} \text{lidar_x} \\ \text{lidar_y} \\ \text{lidar_yaw} \end{bmatrix}$$

2.5 仿真结果

从下图可以看出, Lidar 输出数据波动较大, 经过 EKF 滤波之后效果明显改善, 可以用于控制输入。





3 案例三 （六阶 EKF）

六阶 EKF 主要是用于计算三轴角度，传感器只有 IMU，用于输出俯仰角，滚转角和航向角。核心思想是角速度积分获取角度，加速度通过比例计算当前角度，由于角速度积分获取的角度短时间效果较好，但是长时间会出现偏差，加速度计算的角度噪声较大，但是长时间累计下不会出现偏差，两个互补之下就可以获得很好的角度信息，目前选用六阶 EKF，其中航向角度只是角速度积分获得，由于缺少观测信息，航向角将随时间出现周期变化。

3.1 状态方程

状态向量如下：

$$\text{stateVector} = \begin{bmatrix} q0 \\ q1 \\ q2 \\ q3 \\ \text{dax_b} \\ \text{day_b} \end{bmatrix}$$

其中， $q0, q1, q2, q3$ 是四元数，可转换为三轴角度；

dax_b 代表 roll 角度变化的偏差；

day_b 代表 pitch 角度变化的偏差；

状态更新方程如下：

$$\text{stateVectorNew} = \begin{bmatrix} q0 - (\text{daz}*q3)/2 - q1*(\text{dax}/2 - \text{dax_b}/2) - q2*(\text{day}/2 - \text{day_b}/2) \\ q1 + (\text{daz}*q2)/2 + q0*(\text{dax}/2 - \text{dax_b}/2) - q3*(\text{day}/2 - \text{day_b}/2) \\ q2 - (\text{daz}*q1)/2 + q3*(\text{dax}/2 - \text{dax_b}/2) + q0*(\text{day}/2 - \text{day_b}/2) \\ q3 + (\text{daz}*q0)/2 - q2*(\text{dax}/2 - \text{dax_b}/2) + q1*(\text{day}/2 - \text{day_b}/2) \\ \text{dax_b} \\ \text{day_b} \end{bmatrix}$$

其中， dax 代表 roll 角度变化，角速度积分获得，作为噪声输入；

day 代表 pitch 角度变化，角速度积分获得，作为噪声输入；

daz 代表 yaw 角度变化，角速度积分获得，作为噪声输入；

使用常用公式对上述方程简化：

$$x_{k+1} = F \cdot x_k + G \cdot n_k$$

其中

$$F = \begin{bmatrix} 1, \text{dax_b}/2 - \text{dax}/2, \text{day_b}/2 - \text{day}/2, & -\text{daz}/2, \text{q1}/2, \text{q2}/2 \\ \text{dax}/2 - \text{dax_b}/2, & 1, & \text{daz}/2, \text{day_b}/2 - \text{day}/2, -\text{q0}/2, \text{q3}/2 \\ \text{day}/2 - \text{day_b}/2, & -\text{daz}/2, & 1, \text{dax}/2 - \text{dax_b}/2, -\text{q3}/2, -\text{q0}/2 \\ \text{daz}/2, \text{day}/2 - \text{day_b}/2, \text{dax_b}/2 - \text{dax}/2, & 1, \text{q2}/2, -\text{q1}/2 \\ 0, & 0, & 0, & 0, & 1, & 0 \\ 0, & 0, & 0, & 0, & 0, & 1 \end{bmatrix}$$

$$G = \begin{bmatrix} -\text{q1}/2, -\text{q2}/2, -\text{q3}/2 \\ \text{q0}/2, -\text{q3}/2, \text{q2}/2 \\ \text{q3}/2, \text{q0}/2, -\text{q1}/2 \\ -\text{q2}/2, \text{q1}/2, \text{q0}/2 \\ 0, & 0, & 0 \\ 0, & 0, & 0 \end{bmatrix}, \quad n_k = \begin{bmatrix} \text{dax} \\ \text{day} \\ \text{daz} \end{bmatrix}, \quad \text{三轴角度的变化}$$

$$Q = G \cdot \text{distMatrix} \cdot G^T$$

$$Q = \begin{bmatrix} (\text{daxVar} * \text{q1}^2) / 4 + (\text{dayVar} * \text{q2}^2) / 4 + (\text{dazVar} * \text{q3}^2) / 4 - (\text{daxVar} * \text{q2} * \text{q3}) / 4 - (\text{daxVar} * \text{q1} * \text{q3}) / 4 - (\text{dayVar} * \text{q1} * \text{q2}) / 4 - (\text{daxVar} * \text{q1} * \text{q2}) / 4 - (\text{daxVar} * \text{q0} * \text{q3}) / 4, 0, 0 \\ (\text{dayVar} * \text{q2} * \text{q3}) / 4 - (\text{daxVar} * \text{q0} * \text{q3}) / 4 - (\text{daxVar} * \text{q2} * \text{q3}) / 4 + (\text{dayVar} * \text{q2}^2) / 4 + (\text{daxVar} * \text{q0} * \text{q3}) / 4 - (\text{daxVar} * \text{q1} * \text{q2}) / 4 - (\text{daxVar} * \text{q1} * \text{q3}) / 4 - (\text{daxVar} * \text{q0} * \text{q2}) / 4, 0, 0 \\ (\text{daxVar} * \text{q1} * \text{q3}) / 4 - (\text{dayVar} * \text{q1} * \text{q2}) / 4 - (\text{daxVar} * \text{q1} * \text{q3}) / 4 - (\text{dayVar} * \text{q0} * \text{q3}) / 4 - (\text{daxVar} * \text{q1} * \text{q2}) / 4 - (\text{dayVar} * \text{q0} * \text{q2}) / 4 + (\text{daxVar} * \text{q1}^2) / 4 + (\text{dayVar} * \text{q2}^2) / 4 - (\text{daxVar} * \text{q0} * \text{q1}) / 4, 0, 0 \\ (\text{daxVar} * \text{q1} * \text{q2}) / 4 - (\text{dayVar} * \text{q1} * \text{q2}) / 4 - (\text{daxVar} * \text{q0} * \text{q3}) / 4 - (\text{daxVar} * \text{q0} * \text{q2}) / 4 - (\text{dayVar} * \text{q0} * \text{q3}) / 4 - (\text{daxVar} * \text{q2} * \text{q3}) / 4 - (\text{daxVar} * \text{q0} * \text{q1}) / 4 + (\text{dayVar} * \text{q1}^2) / 4 + (\text{daxVar} * \text{q2}^2) / 4, 0, 0 \\ 0, & 0, & 0, & 0, & 0, & 0 \\ 0, & 0, & 0, & 0, & 0, & 0 \end{bmatrix}$$

$$\text{distMatrix} = \begin{bmatrix} \text{daxVar}, & 0, & 0 \\ 0, \text{dayVar}, & 0 \\ 0, & 0, \text{dazVar} \end{bmatrix} \text{为 } n_k \text{ 噪声方差}$$

其中，daxVar 代表滚转角加速度噪声的方差；

dayVar 代表俯仰角加速度噪声的方差；

dazVar 代表航向角加速度噪声的方差；

上述参数调大，则噪声很大，预测信任度降低，观测信任度提高。

3.2 观测方程

$$z = H \cdot x + R$$

$$z = \begin{bmatrix} \text{ax} \\ \text{ay} \\ \text{az} \end{bmatrix} = \begin{bmatrix} 2 * \text{q1} * \text{q3} - 2 * \text{q0} * \text{q2} \\ 2 * \text{q0} * \text{q1} + 2 * \text{q2} * \text{q3} \\ \text{q0}^2 - \text{q1}^2 - \text{q2}^2 + \text{q3}^2 \end{bmatrix}$$

$$H = H_{\text{ACC}} = \begin{bmatrix} -2 * \text{q2}, & 2 * \text{q3}, & -2 * \text{q0}, & 2 * \text{q1}, & 0, & 0 \\ 2 * \text{q1}, & 2 * \text{q0}, & 2 * \text{q3}, & 2 * \text{q2}, & 0, & 0 \\ 2 * \text{q0}, & -2 * \text{q1}, & -2 * \text{q2}, & 2 * \text{q3}, & 0, & 0 \end{bmatrix}$$

R 是当前时刻观测的不确定度，即 imu 的协方差。

3.3 预测

3.3.1 状态预测

状态预测相对简单，根据状态方程更新状态即可：

$$\text{stateVectorNew} = \begin{bmatrix} q0 - (\text{daz} * q3) / 2 - q1 * (\text{dax} / 2 - \text{dax_b} / 2) - q2 * (\text{day} / 2 - \text{day_b} / 2) \\ q1 + (\text{daz} * q2) / 2 + q0 * (\text{dax} / 2 - \text{dax_b} / 2) - q3 * (\text{day} / 2 - \text{day_b} / 2) \\ q2 - (\text{daz} * q1) / 2 + q3 * (\text{dax} / 2 - \text{dax_b} / 2) + q0 * (\text{day} / 2 - \text{day_b} / 2) \\ q3 + (\text{daz} * q0) / 2 - q2 * (\text{dax} / 2 - \text{dax_b} / 2) + q1 * (\text{day} / 2 - \text{day_b} / 2) \\ \text{dax_b} \\ \text{day_b} \end{bmatrix}$$

3.3.2 协方差预测

协方差预测较为复杂，目前只列出公式：

$$P = F \cdot P \cdot F^T + Q + N_{\text{process}}$$

N_{process} 为除 IMU 噪声以外的滤波器状态过程误差，称过程噪声协方差

$$F = \begin{bmatrix} 1, \text{dax_b}/2 - \text{dax}/2, \text{day_b}/2 - \text{day}/2, & -\text{daz}/2, q1/2, q2/2 \\ \text{dax}/2 - \text{dax_b}/2, & 1, & \text{daz}/2, \text{day_b}/2 - \text{day}/2, -q0/2, q3/2 \\ \text{day}/2 - \text{day_b}/2, & -\text{daz}/2, & 1, \text{dax}/2 - \text{dax_b}/2, -q3/2, -q0/2 \\ \text{daz}/2, \text{day}/2 - \text{day_b}/2, \text{dax_b}/2 - \text{dax}/2, & 1, q2/2, -q1/2 \\ 0, & 0, & 0, & 0, & 1, & 0 \\ 0, & 0, & 0, & 0, & 0, & 1 \end{bmatrix},$$

$$Q = \begin{bmatrix} (\text{daxVar} * q1^2) / 4 + (\text{dayVar} * q2^2) / 4 + (\text{dazVar} * q3^2) / 4 + (\text{daxVar} * q2 * q3) / 4 - (\text{daxVar} * q0 * q1) / 4 - (\text{daxVar} * q2 * q3) / 4, (\text{daxVar} * q1 * q3) / 4 - (\text{dayVar} * q0 * q2) / 4 - (\text{daxVar} * q1 * q3) / 4, (\text{daxVar} * q1 * q2) / 4 - (\text{daxVar} * q0 * q3) / 4, 0, 0 \\ (\text{dayVar} * q2 * q3) / 4 - (\text{daxVar} * q0 * q1) / 4 - (\text{daxVar} * q2 * q3) / 4 + (\text{daxVar} * q0^2) / 4 + (\text{dayVar} * q0^2) / 4 - (\text{daxVar} * q0 * q3) / 4 - (\text{daxVar} * q1 * q3) / 4, (\text{daxVar} * q0 * q3) / 4 - (\text{dayVar} * q0 * q2) / 4 - (\text{daxVar} * q1 * q3) / 4, (\text{daxVar} * q0 * q2) / 4 - (\text{daxVar} * q0 * q3) / 4, 0, 0 \\ (\text{daxVar} * q1 * q3) / 4 - (\text{dayVar} * q0 * q2) / 4 - (\text{daxVar} * q1 * q3) / 4 + (\text{daxVar} * q0^2) / 4 - (\text{daxVar} * q0 * q3) / 4, (\text{dayVar} * q0^2) / 4 + (\text{daxVar} * q1^2) / 4 + (\text{daxVar} * q2^2) / 4, (\text{daxVar} * q0 * q1) / 4 - (\text{daxVar} * q2 * q3) / 4 - (\text{daxVar} * q0 * q1) / 4, 0, 0 \\ (\text{daxVar} * q1 * q2) / 4 - (\text{dayVar} * q0 * q2) / 4 - (\text{daxVar} * q0 * q3) / 4, (\text{daxVar} * q0 * q2) / 4 - (\text{daxVar} * q1 * q3) / 4 - (\text{daxVar} * q0 * q2) / 4, (\text{dayVar} * q0^2) / 4 - (\text{daxVar} * q0 * q1) / 4, (\text{daxVar} * q0^2) / 4 + (\text{dayVar} * q1^2) / 4 + (\text{daxVar} * q2^2) / 4, 0, 0 \\ 0, & 0, & 0, & 0, & 0, & 0 \\ 0, & 0, & 0, & 0, & 0, & 0 \end{bmatrix},$$

$$N_{\text{process}} = [0, 0, 0, 0, \text{dAngBiasSigma}, \text{dAngBiasSigma}]^2$$

3.4 观测

套用以下公式即可：

$$S = H \cdot P_{k+1|k} \cdot H^T + R$$

$$K = P_{k+1|k} \cdot H^T \cdot S^{-1}$$

$$x_{k+1|k+1} = x_{k+1|k} + K \cdot (z - H \cdot x_{k+1|k})$$

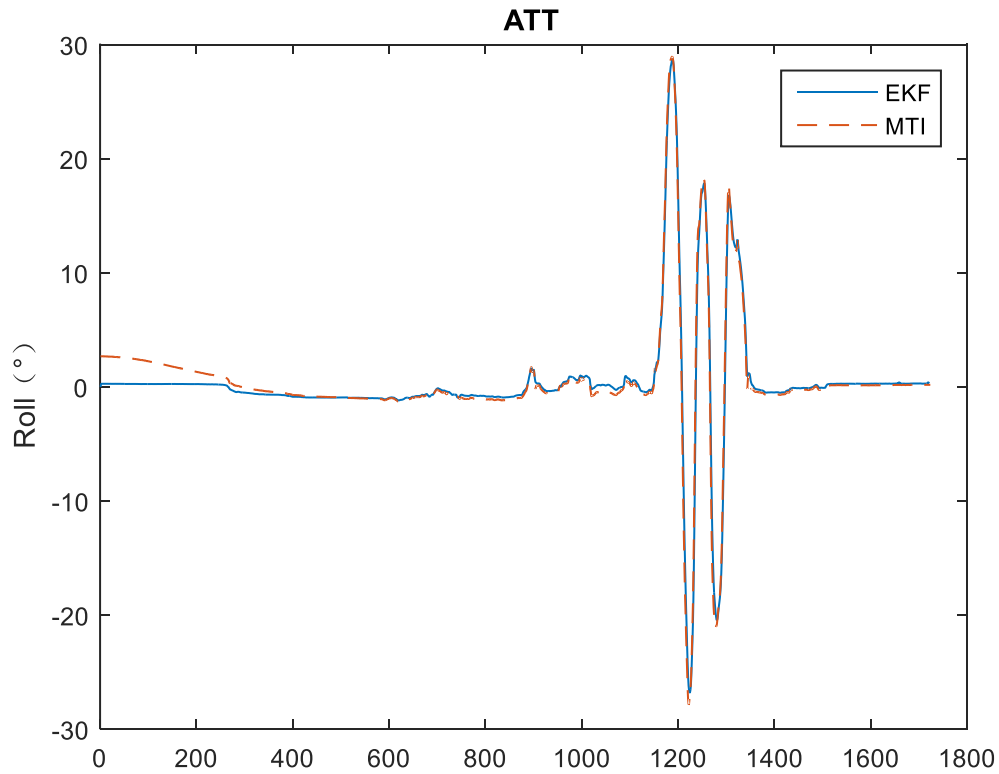
$$P_{k+1|k+1} = [I - K \cdot H] \cdot P_{k+1|k}$$

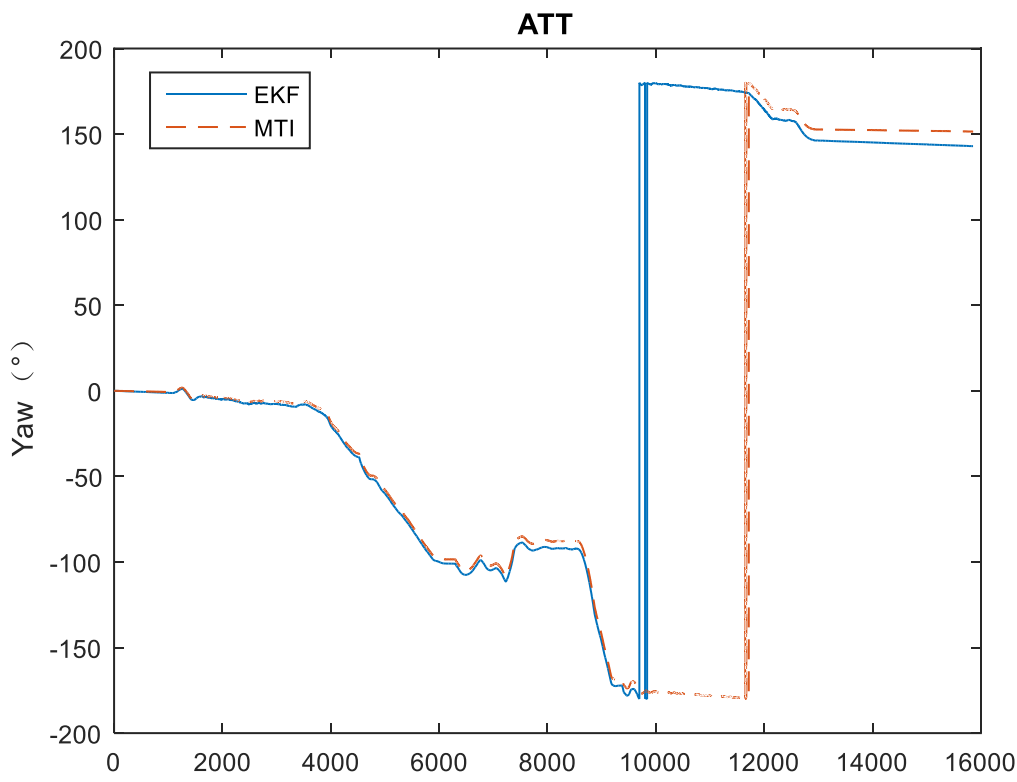
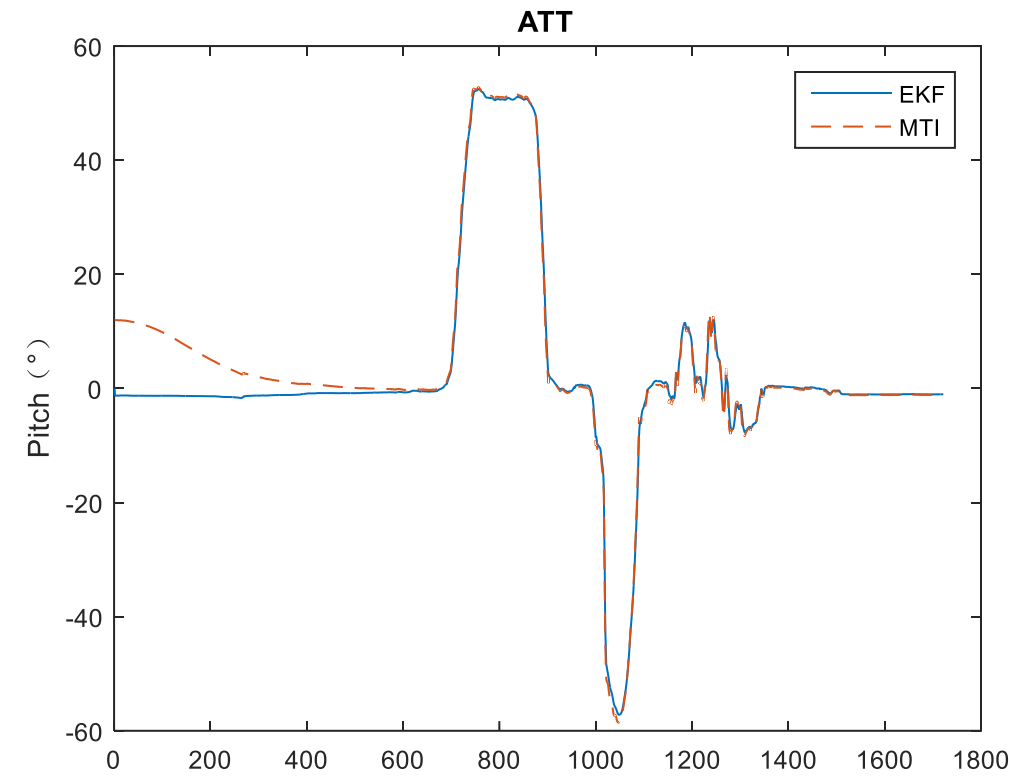
其中，

$$H_{ACC} = \begin{bmatrix} -2*q_2, & 2*q_3, & -2*q_0, & 2*q_1, & 0, & 0 \\ 2*q_1, & 2*q_0, & 2*q_3, & 2*q_2, & 0, & 0 \\ 2*q_0, & -2*q_1, & -2*q_2, & 2*q_3, & 0, & 0 \end{bmatrix}, R = \begin{bmatrix} AccNoise \\ AccNoise \\ AccNoise \end{bmatrix}, z = \begin{bmatrix} ax \\ ay \\ az \end{bmatrix}$$

3.5 仿真结果

下图中，mti 是购买的成品传感器，获取的角度信息算是比较准确的，可以作为参考量，蓝色实线是 EKF 输出数据，从图中可以看出，刚启动时，EKF 初始化之后，角度缓慢收敛；当载体运动时两者的角度变化基本一致，静止之后也基本一致。航向角度由于缺少观测值，短时精度还可以，但是长期精度较差，随着时间累积误差越大。





4 案例四 （十六阶 EKF）

十六阶 EKF 主要是用于划线车导航使用，传感器有 IMU，odo 和差分，用于输出位姿信息，三轴位置，三轴速度，三轴角度，三轴角速度。差分可以提供三轴位置，三轴速度和航向角度信息，但是三轴速度和三轴位置具有一定的延时，运动过程中位置会出现跳点，影响控制精度，odo 可输出前向速度和角速度信息，

4.1 状态方程

状态向量如下：

$$\text{stateVector} = \begin{bmatrix} pn \\ pe \\ pd \\ vn \\ ve \\ vd \\ q0 \\ q1 \\ q2 \\ q3 \\ dax_b \\ day_b \\ daz_b \\ dvx_b \\ dvy_b \\ dvz_b \end{bmatrix}$$

其中， pn 代表正北方向水平位置；

pe 代表正东方向水平位置；

pd 代表指向地面方向水平位置；

vn 代表正北方向水平速度；

ve 代表正东方向水平速度；

vd 代表指向地面方向水平速度；

$q0, q1, q2, q3$ 是四元数，可转换为三轴角度；

dax_b 代表 roll 角度变化的偏差；

day_b 代表 pitch 角度变化的偏差;

daz_b 代表 yaw 角度变化的偏差;

dvx_b 代表正北方向速度变化的偏差;

dvy_b 代表正东方向速度变化的偏差;

dvz_b 代表指向地面方向速度变化的偏差;

状态更新方程如下:

$$\text{stateVectorNew} = \begin{bmatrix} \text{pn} + \text{dt} * \text{vn} \\ \text{pe} + \text{dt} * \text{ve} \\ \text{pd} + \text{dt} * \text{vd} \\ \text{dvx} - \text{dvx_b} + \text{vn} \\ \text{dvy} - \text{dvy_b} + \text{ve} \\ \text{dvz} - \text{dvz_b} + \text{vd} \\ \text{q0} - \text{q1} * (\text{dax}/2 - \text{dax_b}/2) - \text{q2} * (\text{day}/2 - \text{day_b}/2) - \text{q3} * (\text{daz}/2 - \text{daz_b}/2) \\ \text{q1} + \text{q0} * (\text{dax}/2 - \text{dax_b}/2) - \text{q3} * (\text{day}/2 - \text{day_b}/2) + \text{q2} * (\text{daz}/2 - \text{daz_b}/2) \\ \text{q2} + \text{q3} * (\text{dax}/2 - \text{dax_b}/2) + \text{q0} * (\text{day}/2 - \text{day_b}/2) - \text{q1} * (\text{daz}/2 - \text{daz_b}/2) \\ \text{q3} - \text{q2} * (\text{dax}/2 - \text{dax_b}/2) + \text{q1} * (\text{day}/2 - \text{day_b}/2) + \text{q0} * (\text{daz}/2 - \text{daz_b}/2) \\ \text{dax_b} \\ \text{day_b} \\ \text{daz_b} \\ \text{dvx_b} \\ \text{dvy_b} \\ \text{dvz_b} \end{bmatrix}$$

其中, dax 代表 roll 角度变化, 角速度积分获得, 作为噪声输入;

day 代表 pitch 角度变化, 角速度积分获得, 作为噪声输入;

daz 代表 yaw 角度变化, 角速度积分获得, 作为噪声输入;

dvx 代表正北速度变化, odo 坐标系转换获得, 作为噪声输入;

dvy 代表正东速度变化, odo 坐标系转换获得, 作为噪声输入;

dvz 代表指向地面速度变化, odo 坐标系转换获得, 作为噪声输入;

使用常用公式对上述方程简化:

$$x_{k+1} = F \cdot x_k + G \cdot n_k$$

其中

$$F = \begin{bmatrix} 1, 0, 0, dt, 0, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, 0, 0, 0 \\ 0, 1, 0, 0, dt, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, 0, 0, 0 \\ 0, 0, 1, 0, 0, dt, & 0, & 0, & 0, & 0, & 0, & 0, & 0, 0, 0, 0 \\ 0, 0, 0, 1, 0, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, -1, 0, 0 \\ 0, 0, 0, 0, 1, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, 0, -1, 0 \\ 0, 0, 0, 0, 0, 1, & 0, & 0, & 0, & 0, & 0, & 0, & 0, 0, 0, -1 \\ 0, 0, 0, 0, 0, 0, & 1, dax_b/2 - dax/2, day_b/2 - day/2, daz_b/2 - daz/2, q1/2, q2/2, q3/2, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, dax/2 - dax_b/2, & 1, daz/2 - daz_b/2, day_b/2 - day/2, -q0/2, q3/2, -q2/2, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, day/2 - day_b/2, daz_b/2 - daz/2, & 1, dax/2 - dax_b/2, -q3/2, -q0/2, q1/2, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, dax/2 - dax_b/2, day/2 - day_b/2, dax_b/2 - dax/2, & 1, q2/2, -q1/2, -q0/2, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, & 0, & 0, & 0, & 0, & 1, & 0, & 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, & 0, & 0, & 0, & 0, & 0, & 1, & 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 1, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, 1, 0, 0 \\ 0, 0, 0, 0, 0, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, 0, 1, 0 \\ 0, 0, 0, 0, 0, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, 0, 0, 1 \end{bmatrix}$$

$$G = \begin{bmatrix} 0, & 0, & 0, 0, 0, 0 \\ 0, & 0, & 0, 0, 0, 0 \\ 0, & 0, & 0, 0, 0, 0 \\ 0, & 0, & 0, 1, 0, 0 \\ 0, & 0, & 0, 0, 1, 0 \\ 0, & 0, & 0, 0, 0, 1 \\ -q1/2, -q2/2, -q3/2, 0, 0, 0 \\ q0/2, -q3/2, q2/2, 0, 0, 0 \\ q3/2, q0/2, -q1/2, 0, 0, 0 \\ -q2/2, q1/2, q0/2, 0, 0, 0 \\ 0, & 0, & 0, 0, 0, 0 \\ 0, & 0, & 0, 0, 0, 0 \\ 0, & 0, & 0, 0, 0, 0 \\ 0, & 0, & 0, 0, 0, 0 \\ 0, & 0, & 0, 0, 0, 0 \\ 0, & 0, & 0, 0, 0, 0 \end{bmatrix}$$

$$z = \begin{bmatrix} ax \\ ay \\ az \end{bmatrix} = \begin{bmatrix} 2*q1*q3 - 2*q0*q2 \\ 2*q0*q1 + 2*q2*q3 \\ q0^2 - q1^2 - q2^2 + q3^2 \end{bmatrix}$$

$$H=H_ACC = \begin{bmatrix} 0, 0, 0, 0, 0, 0, -2*q2, 2*q3, -2*q0, 2*q1, 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, 2*q1, 2*q0, 2*q3, 2*q2, 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, 2*q0, -2*q1, -2*q2, 2*q3, 0, 0, 0, 0, 0, 0 \end{bmatrix}$$

R 是当前时刻观测的不确定度，即 IMU 的协方差。

4.2.2 差分航向观测

$$z=H \cdot x+R$$

$$z = [\text{gps_yaw}] = [\text{atan2}(2*q0*q3 + 2*q1*q2, q0^2 + q1^2 - q2^2 - q3^2)]$$

$$H=H_GpsYaw = [0, 0, 0, 0, 0, 0, 0, hq1, hq2, hq3, hq4, 0, 0, 0, 0, 0]$$

$$hq1 = (((2*q3)/(q0^2 + q1^2 - q2^2 - q3^2) - (2*q0*(2*q0*q3 + 2*q1*q2))/(q0^2 + q1^2 - q2^2 - q3^2)^2) * (q0^2 + q1^2 - q2^2 - q3^2)^2) / ((2*q0*q3 + 2*q1*q2)^2 + (q0^2 + q1^2 - q2^2 - q3^2)^2)$$

$$hq2 = (((2*q2)/(q0^2 + q1^2 - q2^2 - q3^2) - (2*q1*(2*q0*q3 + 2*q1*q2))/(q0^2 + q1^2 - q2^2 - q3^2)^2) * (q0^2 + q1^2 - q2^2 - q3^2)^2) / ((2*q0*q3 + 2*q1*q2)^2 + (q0^2 + q1^2 - q2^2 - q3^2)^2)$$

$$hq3 = (((2*q1)/(q0^2 + q1^2 - q2^2 - q3^2) + (2*q2*(2*q0*q3 + 2*q1*q2))/(q0^2 + q1^2 - q2^2 - q3^2)^2) * (q0^2 + q1^2 - q2^2 - q3^2)^2) / ((2*q0*q3 + 2*q1*q2)^2 + (q0^2 + q1^2 - q2^2 - q3^2)^2)$$

$$hq4 = (((2*q0)/(q0^2 + q1^2 - q2^2 - q3^2) + (2*q3*(2*q0*q3 + 2*q1*q2))/(q0^2 + q1^2 - q2^2 - q3^2)^2) * (q0^2 + q1^2 - q2^2 - q3^2)^2) / ((2*q0*q3 + 2*q1*q2)^2 + (q0^2 + q1^2 - q2^2 - q3^2)^2)$$

R 是当前时刻观测的不确定度，即 lidar 的协方差。

4.2.1 差分位置观测

$$z=H \cdot x+R$$

$$z = \begin{bmatrix} \text{gps_x} \\ \text{gps_y} \\ \text{gps_z} \end{bmatrix} = \begin{bmatrix} pn \\ pe \\ pd \end{bmatrix}$$

$$H=H_Gps_Pos = \begin{bmatrix} 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 \\ 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 \\ 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 \end{bmatrix}$$

R 是当前时刻观测的不确定度，即 IMU 的协方差。

4.3 预测

4.3.1 状态预测

状态预测相对简单，根据状态方程更新状态即可：

$$\text{stateVectorNew} = \begin{bmatrix} \text{pn} + \text{dt} * \text{vn} \\ \text{pe} + \text{dt} * \text{ve} \\ \text{pd} + \text{dt} * \text{vd} \\ \text{dvx} - \text{dvx_b} + \text{vn} \\ \text{dvy} - \text{dvy_b} + \text{ve} \\ \text{dvz} - \text{dvz_b} + \text{vd} \\ \text{q0} - \text{q1} * (\text{dax}/2 - \text{dax_b}/2) - \text{q2} * (\text{day}/2 - \text{day_b}/2) - \text{q3} * (\text{daz}/2 - \text{daz_b}/2) \\ \text{q1} + \text{q0} * (\text{dax}/2 - \text{dax_b}/2) - \text{q3} * (\text{day}/2 - \text{day_b}/2) + \text{q2} * (\text{daz}/2 - \text{daz_b}/2) \\ \text{q2} + \text{q3} * (\text{dax}/2 - \text{dax_b}/2) + \text{q0} * (\text{day}/2 - \text{day_b}/2) - \text{q1} * (\text{daz}/2 - \text{daz_b}/2) \\ \text{q3} - \text{q2} * (\text{dax}/2 - \text{dax_b}/2) + \text{q1} * (\text{day}/2 - \text{day_b}/2) + \text{q0} * (\text{daz}/2 - \text{daz_b}/2) \\ \text{dax_b} \\ \text{day_b} \\ \text{daz_b} \\ \text{dvx_b} \\ \text{dvy_b} \\ \text{dvz_b} \end{bmatrix}$$

4.3.2 协方差预测

协方差预测较为复杂，目前只列出公式：

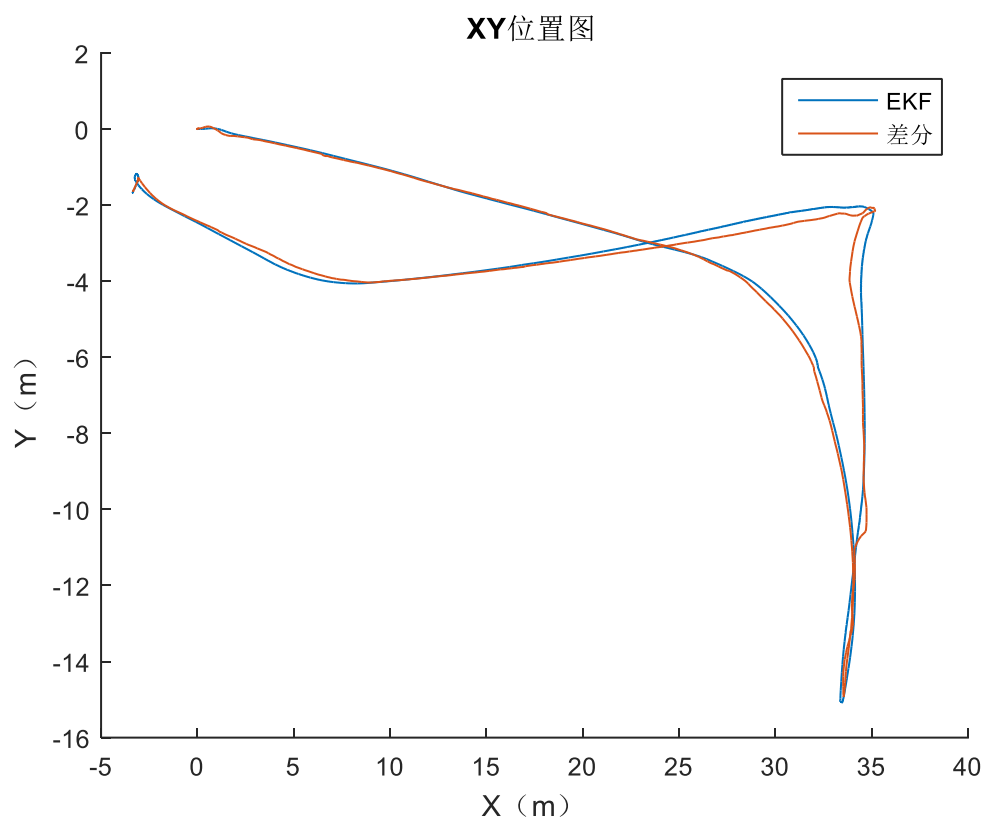
$$\mathbf{P} = \mathbf{F} \cdot \mathbf{P} \cdot \mathbf{F}^T + \mathbf{Q} + \mathbf{N}_{\text{process}}$$

$\mathbf{N}_{\text{process}}$ 为除 IMU 噪声以外的滤波器状态过程误差，称过程噪声协方差

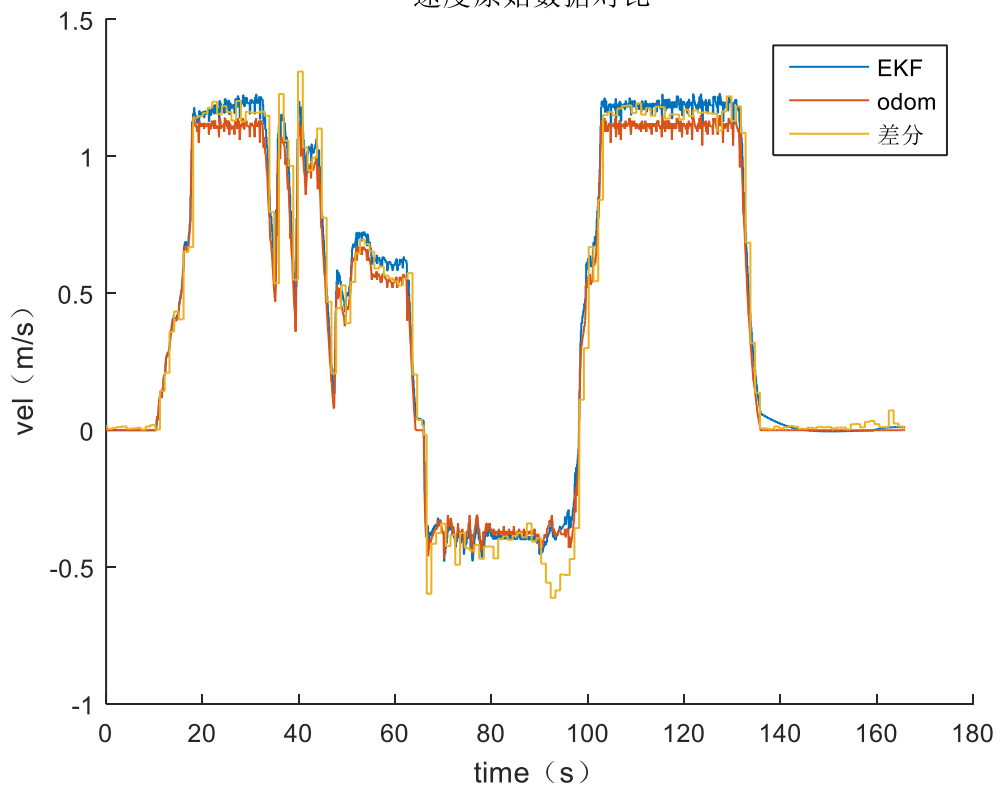
$$\mathbf{F} = \begin{bmatrix} 1, 0, 0, \text{dt}, 0, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0 \\ 0, 1, 0, 0, \text{dt}, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0 \\ 0, 0, 1, 0, 0, \text{dt}, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0 \\ 0, 0, 0, 1, 0, 0, & 0, & 0, & 0, & 0, & 0, & 0, & -1, & 0, & 0 \\ 0, 0, 0, 0, 1, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & -1, & 0 \\ 0, 0, 0, 0, 0, 1, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & -1 \\ 0, 0, 0, 0, 0, 0, & 1, \text{dax_b}/2 - \text{dax}/2, \text{day_b}/2 - \text{day}/2, \text{daz_b}/2 - \text{daz}/2, \text{q1}/2, \text{q2}/2, \text{q3}/2, & 0, & 0, & 0 \\ 0, 0, 0, 0, 0, 0, \text{dax}/2 - \text{dax_b}/2, & 1, \text{daz}/2 - \text{daz_b}/2, \text{day_b}/2 - \text{day}/2, -\text{q0}/2, \text{q3}/2, -\text{q2}/2, & 0, & 0, & 0 \\ 0, 0, 0, 0, 0, 0, \text{day}/2 - \text{day_b}/2, \text{daz_b}/2 - \text{daz}/2, & 1, \text{dax}/2 - \text{dax_b}/2, -\text{q3}/2, -\text{q0}/2, \text{q1}/2, & 0, & 0, & 0 \\ 0, 0, 0, 0, 0, 0, \text{daz}/2 - \text{daz_b}/2, \text{day}/2 - \text{day_b}/2, \text{dax_b}/2 - \text{dax}/2, & 1, \text{q2}/2, -\text{q1}/2, -\text{q0}/2, & 0, & 0, & 0 \\ 0, 0, 0, 0, 0, 0, & 0, & 0, & 0, & 0, & 1, & 0, & 0, & 0, & 0 \\ 0, 0, 0, 0, 0, 0, & 0, & 0, & 0, & 0, & 0, & 1, & 0, & 0, & 0 \\ 0, 0, 0, 0, 0, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 1, & 0, & 0 \\ 0, 0, 0, 0, 0, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 1, & 0 \\ 0, 0, 0, 0, 0, 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0, & 1 \end{bmatrix}$$

4.5 仿真结果

下图中，主要关注的的数据有水平位置，三轴速度和三轴角度，水平位置主要是对比差分数据，从图中可以看出差分数据在转弯时效果不好，不够连续；速度数据中，差分数据呈现阶梯状，但是数值准确性还可以，odo 数据快速性和连续性较好，所以主信 odo。航向数据短时相信角速度积分数据，长期则是相信差分角度。



速度原始数据对比



yaw方向对比

