State Estimation with Extended Kalman Filter

fjctp (snchan20@yahoo.com)

March 26, 2021

Abstract

Estimate Euler angles using extended kalman filter and measurements from IMU (incl. rate gyro, accelometer, and magnetometer).

1 Kalmen Filter

Kalman filter, also known as linear quadratic estimation (LQE), uses a series of measurements (with noise) over time and a mathematical model to estimate unmeasurable variables. The filter involves two steps: predict the unknown variables using a model then update the estimates using measurements.

1.1 Prediction

Use **discrete-time model** to predict unknown variables (x), also known as states, and update the covariance (P) of the estimation.

$$x' = Fx + Bu \tag{1}$$

$$P = FPF^T + Q \tag{2}$$

Note that, Q is process covariance (σ^2). In other word, how good is the model.

1.2 Update

Use sensor measurements (z) to update states (x) and covariance (P).

$$z_{est} = Hx' \tag{3}$$

$$y = z - z_{est} \tag{4}$$

$$S = HPH^T + R \tag{5}$$

$$K = PH^T S^{-1} \tag{6}$$

$$x = x' + Ky \tag{7}$$

$$P = (I - KH)P \tag{8}$$

Note that, R is measurement covariance (σ^2), which is provided by sensor manufacturer or by experiment.

1.3 Extended Kalman Filter

Extended Kalman filter is an extention on Kalman filter, where nonlinear model is used for prediction and update.

Eq (1) becomes

$$x' = f(x, u) \tag{9}$$

Eq (3) becomes

$$z_{est} = h(x') \tag{10}$$

F and H in Eq (2), (5), (6), and (8) becomes Jacobian matrices that are evaluated at x'.

2 Estimate Euler angles

Using extended kalman filter to estimate Euler angles using body angular rate (p, q, r), body acceleration (a_x, a_y, a_z), and body magnetic forces (m_x, m_y, m_z). This section starts with the derivation of the nonlinear model using Kinematic, then it will compute the Jacobian matrics. At the end, it will talk about initialization of the filter and picking Q and R.

2.1 Model

The model is based on kinematic that relates Euler angles, body angular rate, and body acceleration. This section begins with frame rotation, and then the nonlinear kinematic equations. And it will ends with the Jacobian matrices.

2.1.1 Kinematic

Position Given an object's velocity over time, v, we can compute its position, p, by integration.

$$p = \int_0^t v dt \tag{11}$$

In discrete-time, it can be written as

$$p' = p + v\Delta T \tag{12}$$

where ΔT is the time step in second.

Euler angles By integrating angular rate in Euler frame, we will get Euler angles.

$$\begin{bmatrix} \phi' \\ \theta' \\ \psi' \end{bmatrix} = I_{3x3} \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} + R_b^e(\phi, \theta, \psi)\Delta T \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{13}$$

where $R_b^e(\phi, \theta, \psi)$ is a rotation matix that rotates a vector in body frame to euler frame.

2.1.2 Measurement

Body Acceleration Express body acceleration in terms of Euler angles, which is $z_{est} = h_{accel}(x')$ [see Eq. (10)]

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = R_i^b(\phi, \theta, \psi) \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = g \begin{bmatrix} -s_\theta \\ s_\phi c_\theta \\ c_\phi c_\theta \end{bmatrix} \tag{14}$$

where $R_i^b(\phi, \theta, \psi)$ is a rotation matix that rotates a vector in body frame to inertial frame, and g is gravitational acceleration

Compute Jacobian matrix from Eq (14).

$$H_{accel} = \begin{bmatrix} \frac{\partial \dot{u}}{\partial \phi} & \frac{\partial \dot{u}}{\partial \theta} & \frac{\partial \dot{u}}{\partial \psi} \\ \frac{\partial \dot{v}}{\partial \phi} & \frac{\partial \dot{v}}{\partial \theta} & \frac{\partial \dot{v}}{\partial \psi} \\ \frac{\partial \dot{w}}{\partial \phi} & \frac{\partial \dot{w}}{\partial \theta} & \frac{\partial \dot{w}}{\partial \psi} \end{bmatrix} = g \begin{bmatrix} 0 & -c_\theta & 0 \\ -c_\phi c_\theta & -s_\phi s_\theta & 0 \\ -s_\phi c_\theta & -c_\phi s_\theta & 0 \end{bmatrix} \tag{15}$$

Body magnetic forces Express body magnetic forces in terms of Euler angles, which is $z_{est} = h_{mag}(x')$ [see Eq. (10)]

$$\begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} = R_i^b(\phi, \theta, \psi) \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} c_\theta c_\psi \\ -c_\phi s_\psi + s_\phi s_\theta c_\psi \\ s_\phi s_\psi + c_\phi s_\theta c_\psi \end{bmatrix} \tag{16}$$

Note: m_x, m_y, m_z should be normalized, and $R_i^b(\phi, \theta, \psi)$ is a rotation matix that rotates a vector in body frame to inertial frame.

Compute Jacobian matrix from Eq (16).

$$H_{mag} = \begin{bmatrix} 0 & -s_\theta c_\psi & -c_\theta s_\psi \\ s_\phi s_\psi + c_\phi s_\theta c_\psi & s_\phi c_\theta c_\psi & -c_\phi c_\psi - s_\phi s_\theta s_\psi \\ c_\phi s_\psi - s_\phi s_\theta c_\psi & c_\phi c_\theta c_\psi & s_\phi c_\psi - c_\phi s_\theta s_\psi \end{bmatrix} \tag{17}$$

2.1.3 Frame rotation

Rotates acceleration vector to body frame from inertia frame.

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = R_i^b \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \tag{18}$$

where

$$R_i^b = \begin{bmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ (-c_\phi s_\psi + s_\phi s_\theta c_\psi) & (c_\phi c_\psi + s_\phi s_\theta s_\psi) & s_\phi c_\theta \\ (s_\phi s_\psi + c_\phi s_\theta c_\psi) & (-s_\phi c_\psi + c_\phi s_\theta s_\psi) & c_\phi c_\theta \end{bmatrix} \tag{19}$$

Rotates body angular rate vector to Euler frame

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = R_b^e \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{20}$$

where

$$R_b^e = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi / c_\theta & c_\phi / c_\theta \end{bmatrix} \tag{21}$$

2.2 Summary

We want to estimate Euler angles, ϕ, θ, and ψ using Extended Kalman filter. u is body angular rate, p, q, and r. x is Euler angles, ϕ, θ, and ψ. z is body acceleration.

Prediction

$$\begin{bmatrix} \phi' \\ \theta' \\ \psi' \end{bmatrix} = I_{3x3} \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} + R_b^e(\phi, \theta, \psi)\Delta T \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{22}$$

where ΔT is sample time.

$$P = F_j P F_j^T + Q \tag{23}$$

where $F_j = I_{3x3}$

Update

$$z_{est} = H_j x' \tag{24}$$

where measurement matrix, H_j, is a Jacobian matrix.

$$H_j = \begin{bmatrix} \frac{\partial \dot{u}}{\partial \phi} & \frac{\partial \dot{u}}{\partial \theta} & \frac{\partial \dot{u}}{\partial \psi} \\ \frac{\partial \dot{v}}{\partial \phi} & \frac{\partial \dot{v}}{\partial \theta} & \frac{\partial \dot{v}}{\partial \psi} \\ \frac{\partial \dot{w}}{\partial \phi} & \frac{\partial \dot{w}}{\partial \theta} & \frac{\partial \dot{w}}{\partial \psi} \end{bmatrix} = g \begin{bmatrix} 0 & -c_\theta & 0 \\ c_\phi c_\theta & -s_\phi s_\theta & 0 \\ -s_\phi c_\theta & -c_\phi s_\theta & 0 \end{bmatrix} \tag{25}$$

$$y = z - z_{est} \tag{26}$$

$$S = H_j P H_j^T + R \tag{27}$$

$$K = P H_j^T S^{-1} \tag{28}$$

$$x = x' + Ky \tag{29}$$

$$P = (I - KH)P \tag{30}$$

2.3 Initialization

Initialize filter's states, ϕ, θ, ψ using accelerometer measurement, which is body acceleration, a_{bx}, a_{by}, a_{bz}.

Using Equation 14, we have

$$\begin{bmatrix} a_{bx} \\ a_{by} \\ a_{bz} \end{bmatrix} = g \begin{bmatrix} -s_\theta \\ s_\phi c_\theta \\ c_\phi c_\theta \end{bmatrix} \tag{31}$$

We can express ϕ and θ as

$$\theta = -asin(\frac{a_{bx}}{g}) \tag{32}$$

$$\phi = asin(\frac{a_{by}}{g c_\theta}) \tag{33}$$

Since there is no magnetometer, we assume

$$\psi = 0 \tag{34}$$

Assume that the accelerometer is at rest, it returns gravitational acceleration only.

$$g = \sqrt{a_{bx}^2 + a_{by}^2 + a_{bz}^2} \tag{35}$$