

Kalman Filtering & Sensor Fusion

UBC Rocket (TVR, Ulysses)

November 22, 2025

1 Introduction

The goal is to take the data-stream provided from the IMU and through some process determine the current orientation of Ulysses as a quaternion. We have data from the IMU's accelerometer and gyroscope, which we can refer to as follows:

$$\text{Acceleration} = \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix} \quad \text{Angular Velocity} = \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}$$

Which we want to use to find and continuously update $q = q_0 + q_1i + q_2j + q_3k$.

1.1 Notation

Denote $q_{n,k}$ as the prediction of q for time n made at time k . Similarly, we define $a_{x|t}$ and $\omega_{x|t}$ as the value of their respective measurements at time t . Our goal, then, is to find $q_{t,t}$ at our current timestep t , making use of $q_{t,t-1}, q_{t-1,t-1}$, and the measured data. Note that the initial state is $q = 1 + 0i + 0k + 0j$. We will update the state via quaternion multiplication.

1.2 Accelerometer

When Ulysses is stationary or moving at a constant velocity, we expect the magnitude of its acceleration to be approximately equal to 1 (units in g). We can use trigonometry to find the roll/pitch.

$$\begin{aligned} \text{Roll} = \theta_x &= \arctan2(-a_y, -a_z) \\ \text{Pitch} = \theta_y &= \arctan2(a_x, \sqrt{a_y^2 + a_z^2}) \end{aligned}$$

Each of these corresponds to a quaternion rotation:

$$\begin{aligned} q_{\text{roll}} &= \cos(0.5 \cdot \theta_x) + \sin(0.5 \cdot \theta_x)(i + 0j + 0k) \\ q_{\text{pitch}} &= \cos(0.5 \cdot \theta_y) + \sin(0.5 \cdot \theta_y)(0i + j + 0k) \end{aligned}$$

(there is a better way to do this) To update the given state q with any given quaternion rotation r ,

$$q_{\text{new}} = r \cdot q \cdot r^{-1}$$

Where r^{-1} is the conjugate of r . These angles will not be accurate if Ulysses is experiencing acceleration from other sources. We want to fuse this data in order to come to something more accurate regardless of experienced acceleration.

1.3 Gyroscope

By integrating over time with respect to angular velocity, we can find the current angles of Ulysses.

$$q_{t,t} = q_{t-1,t-1} + \frac{\Delta t}{2} \cdot w \cdot q_{t-t,t-1}$$

Where w is the quaternion $w = 0 + \omega_x i + \omega_y j + \omega_z k$. Find the derivation here. We expect the initial values of $\theta_{0,0}$ to be 0, as otherwise have no point of reference to begin our calculations. The key issue with the gyroscope is that it reports a noise 0.1° . Over the course of a flight, this uncertainty will compound, resulting in increasingly erroneous data eventually leading to a spectacular crash. Thus, while the gyroscope data is accurate regardless of acceleration, its noise makes it prone to drift.

2 Filtering

With the setups done, we can begin the process of filtering the data.

2.1 Process

We will begin first with orientation (as we utilize matrix operations, it will be fairly straightforward to combine multiple matrices later on to create an all-in-one orientation/position EKF). The high-level overview is that we predict the next state based on the previous state and the readings from the gyroscope (prediction step), compare the result to the value found from the accelerometer data & decide some way to balance the two results (update step). This is described with the following equations:

$$\hat{x}_k = f(\hat{x}_{k-1}) \quad (1)$$

Where x is the state matrix and $f(x)$ is the state transition.

$$P_{k,k-1} = F_k P_{k-1,k-1} F_k^T + Q \quad (2)$$

Where P is the state covariance matrix, F is the Jacobian matrix of the state transition, and Q is the process noise covariance matrix.

$$F_k = \frac{\partial f(\hat{x}_k)}{\partial \hat{x}_k} \quad (3)$$

Concluding the prediction step. The update step is as follows:

$$\tilde{y}_k = z_k - h(\hat{x}_k) \quad (4)$$

Where \tilde{y}_k is the 'innovation' (difference between prediction and measured), z_k is the measurements of roll & pitch we calculate from the accelerometer (filling 0 in for yaw/heading since we have no way of measuring that accurately), and $h(\hat{x})$ is the observation model previously described (converted into a vector representing the expected gravity/accel readings).

$$K_k = P_{k,k-1} H_k^T (H_k P_{k,k-1} H_k^T + R_k)^{-1} \quad (5)$$

Where K is the Kalman gain, H_k is the Jacobian of $h(\hat{x}_k)$, and R is the measurement covariance matrix (can set to $0.02 \times I$ where I is a 3×3 identity matrix).

$$\hat{x}_{k+1} = \hat{x}_k + K_k \cdot \tilde{y}_k \quad (6)$$

$$P_{k,k} = (I - K_k H_k) P_k \quad (7)$$

We will tune Q and R as necessary. The final expected value that will be fed to the PID is in (6).

2.2 State transition

We first need a way of estimating current orientation based on prior orientation. This is described with (8).

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix} \cdot \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (8)$$

Which is an improvement of the previously described method. \dot{q} is the new quaternion, q is the previous quaternion. Note that this does not rely on the timestep. To arrive at a complete state transition matrix, we must integrate with respect to time, yielding (9).

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} + \frac{\Delta t}{2} \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} \cdot \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (9)$$

2.3 Covariance matrix

We can set it to a 4×4 identity matrix (which we can tune I think).

2.4 State Jacobian

The Jacobian of the state transition is:

$$\begin{bmatrix} 1 & -\omega_x \Delta t & -\omega_y \Delta t & -\omega_z \Delta t \\ \omega_x \Delta t & 1 & \omega_z \Delta t & -\omega_y \Delta t \\ \omega_y \Delta t & -\omega_z \Delta t & 1 & \omega_x \Delta t \\ \omega_z \Delta t & \omega_y \Delta t & -\omega_x \Delta t & 1 \end{bmatrix}$$

2.5 h Jacobian

The Jacobian of $h(x)$ is:

$$\begin{bmatrix} -2q_2 & 2q_3 & -2q_0 & 2q_1 \\ 2q_1 & 2q_0 & 2q_3 & 2q_2 \\ 2q_0 & -2q_1 & -2q_2 & -2q_3 \end{bmatrix}$$

Thus, with everything defined, simply follow the outlined process for a beautiful EKF. Position estimation coming soon!