Quaternion based Extended Kalman Filter as an Attitude and Heading Reference System

Emil Fresk, Luleå University of Technology

August 17, 2012

Abstract

This document goes over the design procedure of a Quaternion based Extended Kalman Filter intended for use as an Attitude and Heading Reference System (AHRS). First quaternion algebra is covered, then the design of the filter and last simulation results. The results were very promising with low noise and small errors.

1 Quaternion algebra

This is a fast recap of quaternion algebra, all can be found in [Diebel, 2006] and [Kuipers, 1998]. A quaternion is a hyper complex number of rank 4. It can be represented in many ways, equation 1 and 2 show two different approaches however equation 2 is what we are going to use.

$$\boldsymbol{q} = q_0 + q_1\boldsymbol{i} + q_2\boldsymbol{j} + q_3\boldsymbol{k} \tag{1}$$

$$\boldsymbol{q} = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}^T \tag{2}$$

1.1 Multiplication

Quaternion multiplication denoted by $\otimes$ can be written as equation 3, 4 and 5. If $\mathbf{p}$ represents one rotation and $\mathbf{q}$ represents another rotation $\mathbf{p} \otimes \mathbf{q}$ represents the combined rotation.

$$\boldsymbol{p} \otimes \boldsymbol{q} = \begin{bmatrix} p_0q_0 - p_1q_1 - p_2q_2 - p_3q_3 \\ p_0q_1 + p_1q_0 + p_2q_3 - p_3q_2 \\ p_0q_2 - p_1q_3 + p_2q_0 + p_3q_1 \\ p_0q_3 + p_1q_2 - p_2q_1 + p_3q_0 \end{bmatrix} \tag{3}$$

$$\boldsymbol{p} \otimes \boldsymbol{q} = Q(\boldsymbol{p})\boldsymbol{q} = \begin{bmatrix} p_0 & -p_1 & -p_2 & -p_3 \\ p_1 & p_0 & -p_3 & p_2 \\ p_2 & p_3 & p_0 & -p_1 \\ p_3 & -p_2 & p_1 & p_0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \tag{4}$$

$$= \bar{Q}(\boldsymbol{q})\boldsymbol{p} = \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & q_3 & -q_2 \\ q_2 & -q_3 & q_0 & q_1 \\ q_3 & q_2 & -q_1 & q_0 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix} \tag{5}$$

1.2 Norm

The norm/length of a quaternion is defined, just as for any complex number, as shown in equation 6. All quaternions in this document are presumed to be of length 1 and are called unit quaternions.

$$\|\boldsymbol{q}\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} \tag{6}$$

1.3 Complex Conjugate

The complex conjugate of a quaternion has the same definition as normal complex numbers. The sign of the complex part is switched as in equation 7.

$$\mathrm{Conj}(\boldsymbol{q}) = \boldsymbol{q}^* = \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \end{bmatrix}^T \tag{7}$$

1.4 Inverse

The inverse of a quaternion is defined as in equation 8, just as the normal inverse of a complex number.

$$\mathrm{Inv}(\boldsymbol{q}) = \boldsymbol{q}^{-1} = \frac{\boldsymbol{q}^*}{\|\boldsymbol{q}\|^2} \tag{8}$$

1.5 Derivative

To find the derivative of a quaternion require some algebraic manipulation but two equations arises. One if the angular velocity vector is in the fixed frame of reference (equation 9) and one if the angular velocity vector is in the body frame of reference (equation 10).

$$\dot{\boldsymbol{q}}_\omega(\boldsymbol{q}, \omega) = \frac{1}{2}\, \boldsymbol{q} \otimes \begin{bmatrix} 0 \\ \omega \end{bmatrix} = \frac{1}{2}\, Q(\boldsymbol{q}) \begin{bmatrix} 0 \\ \omega \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{9}$$

$$\dot{\boldsymbol{q}}_{\omega'}(\boldsymbol{q}, \omega') = \frac{1}{2} \begin{bmatrix} 0 \\ \omega' \end{bmatrix} \otimes \boldsymbol{q} = \frac{1}{2}\, \bar{Q}(\boldsymbol{q}) \begin{bmatrix} 0 \\ \omega' \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & q_3 & -q_2 \\ -q_3 & q_0 & q_1 \\ q_2 & -q_1 & q_0 \end{bmatrix} \begin{bmatrix} \omega'_x \\ \omega'_y \\ \omega'_z \end{bmatrix} \tag{10}$$

1.6 Rotation

If a quaternion is a unit quaternion it can be used as a rotation operator. However the transformation is not built up by only one quaternion multiplication but two as shown in equation 11. This rotates $\mathbf{v}$ from the fixed frame to the body frame represented by $\mathbf{q}$.

$$\boldsymbol{w} = \boldsymbol{q} \otimes \boldsymbol{v} \otimes \boldsymbol{q}^* \tag{11}$$

From this equation we can rewrite it using Q($\mathbf{q}$) formulation resulting in a rotation matrix, rotating a point in a fixed coordinate system, as shown in equation 12 and when rotating a coordinate system the angle sign changes and gives equation 13. The same result arises when conjugating the quaternion in equation 11.

$$R(\boldsymbol{q}) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{12}$$

$$R(\boldsymbol{q})^T = \bar{R}(\boldsymbol{q}) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 + q_0 q_3) & 2(q_1 q_3 - q_0 q_2) \\ 2(q_1 q_2 - q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 + q_0 q_1) \\ 2(q_1 q_3 + q_0 q_2) & 2(q_2 q_3 - q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{13}$$

2 Kalman formulation

A standard implementation of the discrete time Extended Kalman Filter which use continuously changing F and H matrices was used as can be seen in equation 14 to 20.

$$\begin{aligned}
\boldsymbol{x}_k &= f(\boldsymbol{x}_{k-1}, \boldsymbol{u}_k) && (14)\\
P_k &= F_k P_{k-1} F_k^T + Q && (15)\\
\boldsymbol{y}_k &= \boldsymbol{z}_k - h(\boldsymbol{x}_k) && (16)\\
S_k &= H_k P_k H_k^T + R && (17)\\
K_k &= P_k H_k^T S_k^{-1} && (18)\\
\boldsymbol{x}_k &= \boldsymbol{x}_k + K_k \boldsymbol{y}_k && (19)\\
P_k &= (I - K_k H_k) P_k && (20)
\end{aligned}$$

2.1 States

The states were chosen as such: attitude, angular velocity and angular velocity bias, all in the body frame of reference. Translating into one quaternion and two vectors as seen in equation 21.

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{q} & \boldsymbol{\omega} & \boldsymbol{\omega_b} \end{bmatrix}^T = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 & \omega_x & \omega_y & \omega_z & \omega_{xb} & \omega_{yb} & \omega_{zb} \end{bmatrix}^T \quad (21)$$

2.2 Prediction

When calculating the prediction of the quaternion in discrete time it's as simple as to numerically integrate the derivative of the quaternion from equation 10 as shown in equation 22.

$$\boldsymbol{q}_k = \boldsymbol{q}_{k-1} + \Delta t \cdot \dot{\boldsymbol{q}}_{\omega'} \quad (22)$$

Knowing this, the non-linear state space equation arises as in equation 23. We assume that the angular velocity and angular velocity bias does not change very much from each sample to the next. If more exact estimation was needed, one could add the effect of control signal to this, however this is omitted in this implementation.

$$f(\boldsymbol{x}, \boldsymbol{u}) = \begin{bmatrix}
q_0 - \frac{\Delta t}{2}(q_1\omega_x + q_2\omega_y + q_3\omega_z) \\
q_1 + \frac{\Delta t}{2}(q_0\omega_x + q_3\omega_y - q_2\omega_z) \\
q_2 - \frac{\Delta t}{2}(q_3\omega_x - q_0\omega_y - q_1\omega_z) \\
q_3 + \frac{\Delta t}{2}(q_2\omega_x - q_1\omega_y + q_0\omega_z) \\
\omega_x \\
\omega_y \\
\omega_z \\
\omega_{xb} \\
\omega_{yb} \\
\omega_{zb}
\end{bmatrix} \quad (23)$$

And taking the Jacobian of this equation with respect to the state gives the F matrix as in equation 24.

$$\frac{\partial f}{\partial \boldsymbol{x}} = F_k = \begin{bmatrix}
1 & -\frac{\Delta t}{2}\omega_x & -\frac{\Delta t}{2}\omega_y & -\frac{\Delta t}{2}\omega_z & -\frac{\Delta t}{2}q_1 & -\frac{\Delta t}{2}q_2 & -\frac{\Delta t}{2}q_3 & 0 & 0 & 0 \\
\frac{\Delta t}{2}\omega_x & 1 & -\frac{\Delta t}{2}\omega_z & \frac{\Delta t}{2}\omega_y & \frac{\Delta t}{2}q_0 & \frac{\Delta t}{2}q_3 & -\frac{\Delta t}{2}q_2 & 0 & 0 & 0 \\
\frac{\Delta t}{2}\omega_y & \frac{\Delta t}{2}\omega_z & 1 & -\frac{\Delta t}{2}\omega_x & -\frac{\Delta t}{2}q_3 & \frac{\Delta t}{2}q_0 & \frac{\Delta t}{2}q_1 & 0 & 0 & 0 \\
\frac{\Delta t}{2}\omega_z & -\frac{\Delta t}{2}\omega_y & \frac{\Delta t}{2}\omega_x & 1 & \frac{\Delta t}{2}q_2 & -\frac{\Delta t}{2}q_1 & \frac{\Delta t}{2}q_0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix} \quad (24)$$

2.3 Measurements

There are three sensors in the system that will be accounted for; the accelerometer, gyroscope and magnetometer. The accelerometer and magnetometer are used to find the attitude, heading and gyroscope bias while the gyroscope is used to for fast dynamic changes. This gives the measurement vector as shown in equation 25. Measurements from the accelerometer and magnetometer are chosen to be normalized for the direction is only needed. $\boldsymbol{\omega_m}$ is in rad/s.

$$\boldsymbol{z}(k) = \begin{bmatrix} \boldsymbol{a} & \boldsymbol{\omega_m} & \boldsymbol{m} \end{bmatrix}^T = \begin{bmatrix} a_x & a_y & a_z & \omega_{xm} & \omega_{ym} & \omega_{zm} & m_x & m_y & m_z \end{bmatrix}^T \tag{25}$$

2.4 Mapping of measurements to states

In order to get the measurements to align with the states the connection between measurements and states must be made. This is done by finding the non-linear equation $h(\boldsymbol{x})$ and its Jacobian. The non-linear equation is used for calculating the error and the Jacobian is for calculating the Kalman gain and updating the covariance matrix.

2.4.1 Acceleration measurements

First let the normalized gravity vector seen in equation 26 be a fixed vector in space and rotate it's coordinate frame into the measurement's frame by using the estimated attitude. From the non-linear vector equation that arises the Jacobian with respect to state can be calculated as shown in equation 28. Thus the mapping of the accelerometer measurements is complete.

$$\boldsymbol{g} = \begin{bmatrix} 0 & 0 & -1 \end{bmatrix}^T \tag{26}$$

$$\boldsymbol{a} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \bar{R}(\boldsymbol{q}) \cdot \boldsymbol{g} = \begin{bmatrix} -2(q_1 q_3 - q_0 q_2) \\ -2(q_2 q_3 + q_0 q_1) \\ -q_0^2 + q_1^2 + q_2^2 - q_3^2 \end{bmatrix} \tag{27}$$

$$\frac{\partial \boldsymbol{a}}{\partial \boldsymbol{q}} = J_a(\boldsymbol{q}) = \begin{bmatrix} 2q_2 & -2q_3 & 2q_0 & -2q_1 \\ -2q_1 & -2q_0 & -2q_3 & -2q_2 \\ -2q_0 & 2q_1 & 2q_2 & -2q_3 \end{bmatrix} \tag{28}$$

2.4.2 Gyroscopic measurements

The gyroscopic mapping is much easier to find. As the measurement and state is in the same frame of reference no rotation is needed. This results in two identity matrices as shown in equation 29 and thus the mapping of the gyroscopic measurements is complete.

$$\boldsymbol{\omega_m} = \begin{bmatrix} \omega_{xm} \\ \omega_{ym} \\ \omega_{zm} \end{bmatrix} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} + \begin{bmatrix} \omega_{xb} \\ \omega_{yb} \\ \omega_{zb} \end{bmatrix} = \begin{bmatrix} H_\omega & H_\omega \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{\omega_b} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \\ \omega_{xb} \\ \omega_{yb} \\ \omega_{zb} \end{bmatrix} \tag{29}$$

2.4.3 Magnetic measurements

The same procedure as with the accelerometer can be used here, just that the fixed vector is now the earth's magnetic field as shown in equation 30.

$$\boldsymbol{b} = \begin{bmatrix} b_x & 0 & b_z \end{bmatrix}^T \tag{30}$$

In order to find b_x and b_z we use our knowledge of the attitude of the system to calculate them. First we rotate the measurement to the earth frame of reference as in equation 31. When the coordinate systems are aligned it's as simple as identifying $b_z = h_z$ and calculate the magnitude of b_x as in equation 32. Magnetic disturbances will be limited to only affect the estimated heading with this formulation

[Madgwick, 2010]. It's important to note that the east/west direction of the field is zero and it only has a vertical and a north component.

$$\boldsymbol{h} = \begin{bmatrix} h_x & h_y & h_z \end{bmatrix}^T = \boldsymbol{m} \cdot R(\boldsymbol{q}_{est}) \tag{31}$$

$$\boldsymbol{b} = \begin{bmatrix} b_x & 0 & b_z \end{bmatrix}^T = \begin{bmatrix} \sqrt{h_x^2 + h_y^2} & 0 & h_z \end{bmatrix}^T \tag{32}$$

Now rotate the magnetic vector's coordinate frame into the measurement's frame. Here we can also calculate the Jacobian of the non-linear vector equation, as shown in equation 34.

$$\boldsymbol{m} = \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} = \bar{R}(\boldsymbol{q}) \cdot \boldsymbol{b} = \begin{bmatrix} (q_0^2 + q_1^2 - q_2^2 - q_3^2)b_x + 2(q_1q_3 - q_0q_2)b_z \\ 2(q_1q_2 - q_0q_3)b_x + 2(q_2q_3 + q_0q_1)b_z \\ 2(q_1q_3 + q_0q_2)b_x + (q_0^2 - q_1^2 - q_2^2 + q_3^2)b_z \end{bmatrix} \tag{33}$$

$$\frac{\partial \boldsymbol{m}}{\partial \boldsymbol{q}} = J_m(\boldsymbol{q}) = \begin{bmatrix} 2(q_0b_x - q_2b_z) & 2(q_1b_x + q_3b_z) & -2(q_2b_x + q_0b_z) & -2(q_3b_x - q_1b_z) \\ -2(q_3b_x - q_1b_z) & 2(q_2b_x + q_0b_z) & 2(q_1b_x + q_3b_z) & -2(q_0b_x - q_2b_z) \\ 2(q_2b_x + q_0b_z) & 2(q_3b_x - q_1b_z) & 2(q_0b_x - q_2b_z) & 2(q_1b_x + q_3b_z) \end{bmatrix} \tag{34}$$

2.4.4 h(x) and the complete H matrix

Combining equation 27, 29 and 33 gives the non-linear equation $h(\boldsymbol{x})$, from equation 16, as seen in equation 35.

$$h(\boldsymbol{x}) = \begin{bmatrix} -2(q_1q_3 - q_0q_2) \\ -2(q_2q_3 + q_0q_1) \\ -q_0^2 + q_1^2 + q_2^2 - q_3^2 \\ \omega_x + \omega_{xb} \\ \omega_y + \omega_{yb} \\ \omega_z + \omega_{zb} \\ (q_0^2 + q_1^2 - q_2^2 - q_3^2)b_x + 2(q_1q_3 - q_0q_2)b_z \\ 2(q_1q_2 - q_0q_3)b_x + 2(q_2q_3 + q_0q_1)b_z \\ 2(q_1q_3 + q_0q_2)b_x + (q_0^2 - q_1^2 - q_2^2 + q_3^2)b_z \end{bmatrix} \tag{35}$$

The H matrix is now built by the matrices before as shown in equation 36 and 37. This matrix is the same as calculating the Jacobian of equation 35.

$$\frac{\partial h}{\partial \boldsymbol{x}} = H_k = \begin{bmatrix} J_a(\boldsymbol{q}) & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & H_\omega & H_\omega \\ J_m(\boldsymbol{q}) & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} = \tag{36}$$

$$= \begin{bmatrix} 2q_2 & -2q_3 & 2q_0 & -2q_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2q_1 & -2q_0 & -2q_3 & -2q_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2q_0 & 2q_1 & 2q_2 & -2q_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 2(q_0b_x - q_2b_z) & 2(q_1b_x + q_3b_z) & -2(q_2b_x + q_0b_z) & -2(q_3b_x - q_1b_z) & 0 & 0 & 0 & 0 & 0 & 0 \\ -2(q_3b_x - q_1b_z) & 2(q_2b_x + q_0b_z) & 2(q_1b_x + q_3b_z) & -2(q_0b_x - q_2b_z) & 0 & 0 & 0 & 0 & 0 & 0 \\ 2(q_2b_x + q_0b_z) & 2(q_3b_x - q_1b_z) & 2(q_0b_x - q_2b_z) & 2(q_1b_x + q_3b_z) & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{37}$$

2.4.5 Quaternion re-normalization

It's important no note that if the quaternion part of the state, in equation 21, does not have a length of one it will no longer represent a rotation and numerical integration and corrections are bound to violate this constraint. Therefore it's important to re-normalize the quaternion after every time a change has been made to it (equation 14 and 19) by dividing it with its norm from equation 6 as seen in equation 38.

$$\boldsymbol{q}_{norm} = \frac{\boldsymbol{q}}{\|\boldsymbol{q}\|} \tag{38}$$

3 Simulation

When simulating the system Simulink in Matlab was used. The Simulink model can be seen in figure 1 and the choice of the tuning matrices Q and R can be seen in equation 39 and 40.

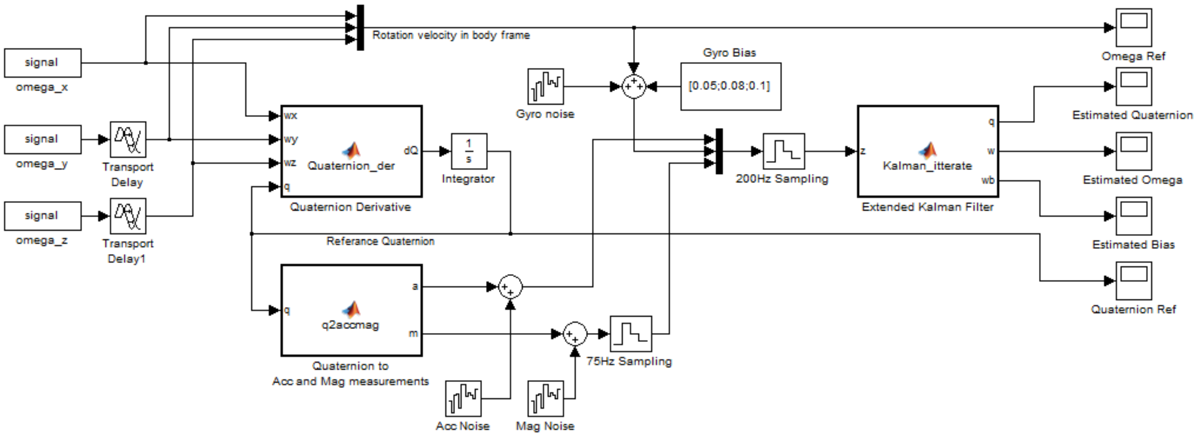

Figure 1: Simulink model.

$$Q = \begin{bmatrix} 5 & & & & & & & & \cdots & 0 \\ & 5 & & & & & & & & \vdots \\ & & 5 & & & & & & & \\ & & & 5 & & & & & & \\ & & & & 10^3 & & & & & \\ & & & & & 10^3 & & & & \\ & & & & & & 10^3 & & & \\ & & & & & & & 0.1 & & \\ \vdots & & & & & & & & 0.1 & \\ 0 & \cdots & & & & & & & & 0.1 \end{bmatrix} \tag{39}$$

$$R = \begin{bmatrix} 10^6 & & & & & & & & \cdots & 0 \\ & 10^6 & & & & & & & & \vdots \\ & & 10^6 & & & & & & & \\ & & & 10^4 & & & & & & \\ & & & & 10^4 & & & & & \\ & & & & & 10^4 & & & & \\ & & & & & & 10^6 & & & \\ \vdots & & & & & & & 10^6 & & \\ 0 & \cdots & & & & & & & 10^6 & \end{bmatrix} \tag{40}$$

3.1 Simulation Results

The results are promising. The filter follows the references well even though large amounts of noise are added and different sampling intervals for different sensors are used. The biggest error is in the angular velocity, but it's still so small that it doesn't effect the rest of the system. The tracking and errors can be seen in figure 2 to 5. The bias estimation tracks very well. It shows small deviations as expected but it holds it value very well. The bias error can be seen in figure 6.

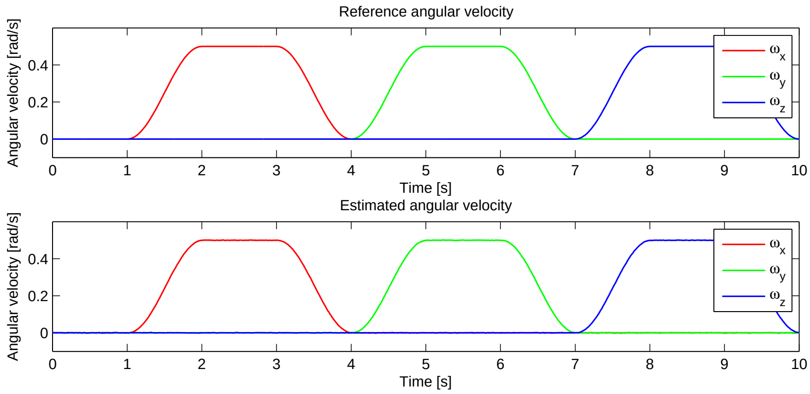

Figure 2: Reference angular velocity compared to estimated angular velocity.

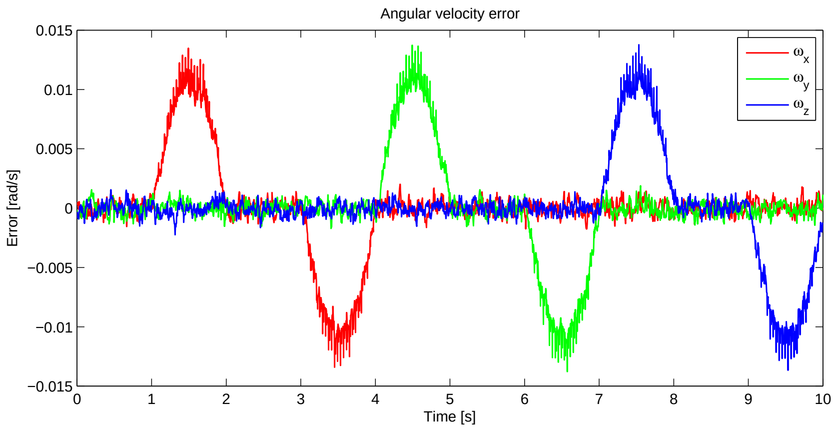

Figure 3: Angular velocity error.

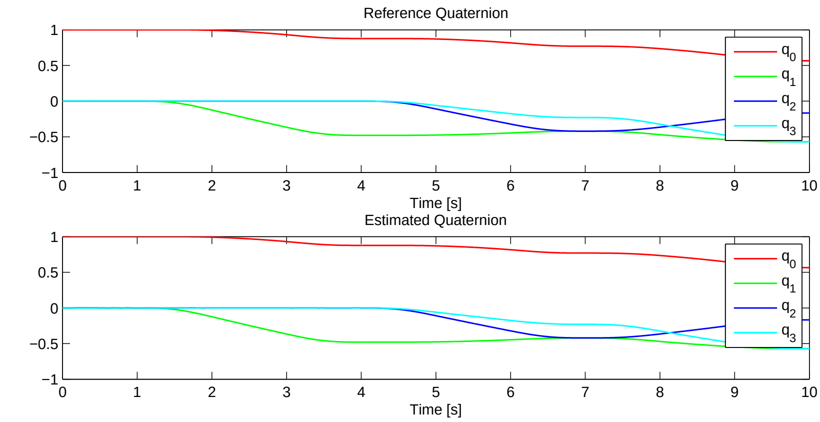

Figure 4: Reference quaternion compared to estimated quaternion.

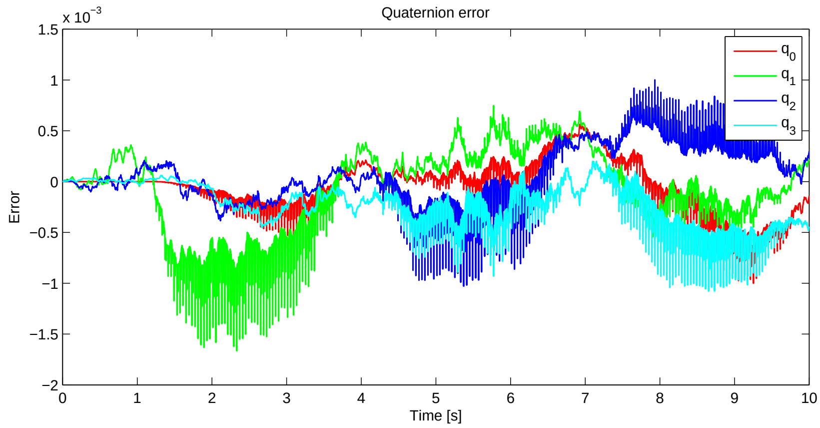

Figure 5: Quaternion error.

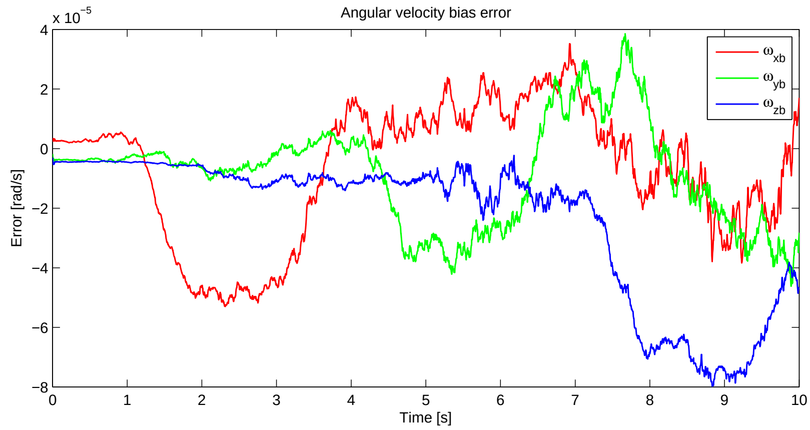

Figure 6: Bias error.

References

[Diebel, 2006] Diebel, J. (2006). Representing attitude: Euler angles, unit quaternions, and rotation vectors.

[Kuipers, 1998] Kuipers, J. B. (1998). *Quaternions and Rotation Sequences.*

[Madgwick, 2010] Madgwick, S. O. (2010). An efficient orientation filter for inertial and inertial/magnetic sensor arrays.