Libxmotion Implementation Notes

Ruixiang Du
ruixiang.du@gmail.com

May 11, 2024

Contents

1 MEKF

This section is about the MEKF implementation in libxmotion. Most of the equations and derivation steps are taken from [1].

1.1 Quaternion

The definition of a quaternion is given by:

$$\boldsymbol{q} = \begin{bmatrix} q_1 \\ \boldsymbol{q}_{2:4} \end{bmatrix} \tag{1}$$

The relevant operations on quaternions that are used for MEKF are given as follows:

- Multiplication:

$$\boldsymbol{q} \otimes \boldsymbol{p} = \begin{bmatrix} q_1 p_1 - \boldsymbol{q}_{2:4}\boldsymbol{p}_{2:4} \\ q_1 \boldsymbol{p}_{2:4} + p_1 \boldsymbol{q}_{2:4} + \boldsymbol{q}_{2:4} \times \boldsymbol{p}_{2:4} \end{bmatrix} \tag{2}$$

- Inverse:

$$\boldsymbol{q}^{-1} = \begin{bmatrix} q_1 \\ -\boldsymbol{q}_{2:4} \end{bmatrix} \tag{3}$$

A quaternion represents a rotation of one frame with respect to another. The inverse of a quaternion represents the rotation in the opposite direction. The multiplication of two quaternions represents the composition of two rotations.

We have the following properties of quaternions:

$$\boldsymbol{q} \otimes \boldsymbol{q}^{-1} = \begin{bmatrix} 1 \\ \boldsymbol{0} \end{bmatrix} \tag{4}$$

$$\begin{aligned} \dot{\boldsymbol{q}} &= \frac{1}{2} \boldsymbol{q} \otimes \begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} \\ &= \frac{1}{2} \boldsymbol{\Omega}(\boldsymbol{\omega}) \boldsymbol{q} \end{aligned} \tag{5}$$

where ω is the angular velocity vector of the body frame with respect to the inertial frame and Ω is defined as

$$\boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\boldsymbol{\omega} \\ \boldsymbol{\omega} & -[\boldsymbol{\omega}_\times] \end{bmatrix} \tag{6}$$

$$\boldsymbol{\omega}_\times = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{7}$$

1.2 Error State Model

The MEKF is used to estimate the attitude of the body frame with respect to the inertial frame. The main difference between MEKF and other Kalman filters for attitude estimation is that MEKF does not use the quaternion in the state vector directly.

Normally, we would have the state dynamics as

$$\dot{\hat{\boldsymbol{q}}} = \frac{1}{2}\hat{\boldsymbol{q}} \otimes \begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} \tag{8}$$

$$\dot{\hat{\boldsymbol{v}}}^i = \boldsymbol{C}_b^i(\hat{\boldsymbol{q}})\hat{\boldsymbol{f}}^b + \hat{\boldsymbol{g}}^i \tag{9}$$

$$\dot{\hat{\boldsymbol{r}}}^i = \hat{\boldsymbol{v}}^i \tag{10}$$

where $\boldsymbol{C}_b^i$ is the transformation matrix that transforms a vector from the body frame to the inertial frame, $\hat{\boldsymbol{q}}$ is the estimated quaternion, $\hat{\boldsymbol{v}}$ is the estimated velocity, and $\hat{\boldsymbol{r}}$ is the estimated position.

With MEKF, we use the error quaternion, which is the difference between the estimated quaternion and the true quaternion, as well as error velocity and position.

The error quaternion is given as

$$\boldsymbol{q} = \hat{\boldsymbol{q}} \otimes \delta\boldsymbol{q} \tag{11}$$

$$\Rightarrow \delta\boldsymbol{q} = \hat{\boldsymbol{q}}^{-1} \otimes \boldsymbol{q} \tag{12}$$

where $\hat{\boldsymbol{q}}$ is the estimated quaternion and $\delta\boldsymbol{q}$ is the error quaternion.

The error state vector is then defined as

$$\delta\boldsymbol{x} = \begin{bmatrix} \delta\boldsymbol{q} \\ \delta\boldsymbol{v} \\ \delta\boldsymbol{r} \\ \boldsymbol{\beta}_{\boldsymbol{\omega}} \\ \boldsymbol{\beta}_{\boldsymbol{f}} \\ \boldsymbol{\beta}_{\boldsymbol{m}} \end{bmatrix} \tag{13}$$

where $\delta\boldsymbol{q}$ is the error quaternion, $\delta\boldsymbol{v}$ is the error in velocity, $\delta\boldsymbol{r}$ is the error in position, $\boldsymbol{\beta}_{\boldsymbol{\omega}}$ is the bias in the angular velocity, $\boldsymbol{\beta}_{\boldsymbol{f}}$ is the bias in the specific force, and $\boldsymbol{\beta}_{\boldsymbol{m}}$ is the bias in the magnetometer.

Then we need to derive the dynamics of the error state $\dot{\boldsymbol{x}}$. We can do it by examining each component of the state vector separately.

1.2.1 Error Quaternion Dynamics

The error quaternion dynamics is derived as follows:

$$\begin{gathered} \delta\boldsymbol{q} = \hat{\boldsymbol{q}}^{-1} \otimes \boldsymbol{q} \\ \Rightarrow \delta\dot{\boldsymbol{q}} = \hat{\boldsymbol{q}}^{-1} \otimes \dot{\boldsymbol{q}} + \dot{\hat{\boldsymbol{q}}}^{-1} \otimes \boldsymbol{q} \end{gathered} \tag{14}$$

After the following steps described in [1], we will eventually get

$$\delta\dot{\boldsymbol{q}}_{2:4} \cong -\hat{\boldsymbol{\omega}}_{\times}\delta\boldsymbol{q}_{2:4} + \frac{1}{2}\delta\boldsymbol{\omega} \tag{15}$$

with the fact that $\delta q_1 = 1$ and the assumption that $\delta\boldsymbol{q}$ is small.

Here we replace the error states $\delta\boldsymbol{q}_{\{2:4\}}$ with a vector of small angles to further simplify the equations.

$$\begin{gathered} \boldsymbol{\alpha} = 2\delta\boldsymbol{q}_{2:4} \\ \Rightarrow \dot{\boldsymbol{\alpha}} = -\hat{\boldsymbol{\omega}}_{\times}\boldsymbol{\alpha} + \delta\boldsymbol{\omega} \end{gathered} \tag{16}$$

1.2.2 Error Velocity and Position Dynamics

For the velocity and position error, we have

$$\delta \boldsymbol{v} = \boldsymbol{v} - \hat{\boldsymbol{v}} \tag{17}$$
$$\delta \boldsymbol{r} = \boldsymbol{r} - \hat{\boldsymbol{r}} \tag{18}$$

Based on Equation 9 and Equation 10, we can derive the dynamics of the velocity and position error as

$$\delta \dot{\boldsymbol{v}} = -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}})\hat{\boldsymbol{f}}_\times^b \boldsymbol{\alpha} + \boldsymbol{C}_b^i \delta \boldsymbol{f} \tag{19}$$
$$\delta \dot{\boldsymbol{r}} = \delta \boldsymbol{v} \tag{20}$$

1.2.3 Sensor Bias Dynamics

The angular rate error model is given by

$$\boldsymbol{\omega} = \hat{\boldsymbol{\omega}} - \boldsymbol{\beta}_\omega - \boldsymbol{\eta}_\omega \tag{21}$$
$$\dot{\boldsymbol{\beta}}_\omega = \boldsymbol{\nu}_\omega \tag{22}$$

Together with the definition

$$\boldsymbol{\omega} = \hat{\boldsymbol{\omega}} + \delta \boldsymbol{\omega} \tag{23}$$

We can get

$$\delta \boldsymbol{\omega} = -\boldsymbol{\beta}_\omega - \boldsymbol{\eta}_\omega \tag{24}$$
$$\delta \dot{\omega} = -\boldsymbol{\nu}_\omega \tag{25}$$

The linear acceleration error model is given by

$$\boldsymbol{f} = \hat{\boldsymbol{f}} - \boldsymbol{\beta}_f - \boldsymbol{\eta}_f \tag{26}$$
$$\dot{\boldsymbol{\beta}}_f = \boldsymbol{\nu}_f \tag{27}$$

Together with the definition

$$\boldsymbol{f} = \hat{\boldsymbol{f}} + \delta \boldsymbol{f} \tag{28}$$

We can get

$$\delta \boldsymbol{f} = -\boldsymbol{\beta}_f - \boldsymbol{\eta}_f \tag{29}$$

The magnetometer bias is treated as a slowly diverging random walk process driven by the noise process $\boldsymbol{\nu}_m$

$$\dot{\boldsymbol{\beta}}_m = \boldsymbol{\nu}_m \tag{30}$$

1.2.4 Full Error State Dynamics

According to the results above, we redefine the error state as

$$
\delta \boldsymbol{x} = \begin{bmatrix} \boldsymbol{\alpha} \\ \delta \boldsymbol{v} \\ \delta \boldsymbol{r} \\ \boldsymbol{\beta_\omega} \\ \boldsymbol{\beta_f} \\ \boldsymbol{\beta_m} \end{bmatrix} \tag{31}
$$

Combining equation Equation 16 Equation 19 Equation 20 Equation 25 Equation 27 Equation 30, together with Equation 24 and Equation 29, we get the full dynamics of the error state:

$$
\delta \dot{\boldsymbol{x}} = \begin{bmatrix} \dot{\boldsymbol{\alpha}} \\ \delta \dot{\boldsymbol{v}} \\ \delta \dot{\boldsymbol{r}} \\ \dot{\boldsymbol{\beta}}_{\boldsymbol{\omega}} \\ \dot{\boldsymbol{\beta}}_{\boldsymbol{f}} \\ \dot{\boldsymbol{\beta}}_{\boldsymbol{m}} \end{bmatrix} = \begin{bmatrix} -\hat{\boldsymbol{\omega}}_\times & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}})\hat{\boldsymbol{f}}^b_\times & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}}) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \delta \boldsymbol{v} \\ \delta \boldsymbol{r} \\ \boldsymbol{\beta_\omega} \\ \boldsymbol{\beta_f} \\ \boldsymbol{\beta_m} \end{bmatrix} + \begin{bmatrix} -\boldsymbol{\eta_\omega} \\ -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}})\boldsymbol{\eta}_f \\ \mathbf{0}_{3\times3} \\ \boldsymbol{\nu}_\omega \\ \boldsymbol{\nu}_f \\ \boldsymbol{\nu}_m \end{bmatrix} \tag{32}
$$

The 18-error-state model is linear and time-varying with respect to the error states. Following the standard representation of a linear time-varying system, we can write the error state dynamics as

$$
\delta \dot{\boldsymbol{x}} = \boldsymbol{F} \delta \boldsymbol{x} + \boldsymbol{G} \mathtt{w} \tag{33}
$$

where matrices $\boldsymbol{F}$ is given by

$$
\boldsymbol{F}\left(\hat{\boldsymbol{q}}, \hat{\boldsymbol{\omega}}, \hat{\boldsymbol{f}}\right) = \begin{bmatrix} -\hat{\boldsymbol{\omega}}_\times & \mathbf{0} & \mathbf{0} & -\boldsymbol{I}_{3\times3} & \mathbf{0} & \mathbf{0} \\ -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}})\hat{\boldsymbol{f}}^b_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}}) & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_{3\times3} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{34}
$$

the matrix $\boldsymbol{G}$ transforms the white noice sequeunce $\mathtt{w}$ into the disturbance vector and is given by

$$
\boldsymbol{G} = \begin{bmatrix} -\boldsymbol{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}}) & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I} \end{bmatrix} \tag{35}
$$

The variance of the disturbance vector $\mathtt{w}$ is given by

$$
\boldsymbol{Q}_c = \begin{bmatrix}
\operatorname{diag}(\boldsymbol{\sigma}_\omega^2) & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \operatorname{diag}\left(\boldsymbol{\sigma}_f^2\right) & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \operatorname{diag}\left(\boldsymbol{\sigma}_{\beta\omega}^2\right) & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \operatorname{diag}\left(\boldsymbol{\sigma}_{\beta f}^2\right) & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \operatorname{diag}\left(\boldsymbol{\sigma}_{\beta m}^2\right)
\end{bmatrix} \tag{36}
$$

where $\boldsymbol{\sigma}_\omega$, $\boldsymbol{\sigma}_f$, $\boldsymbol{\sigma}_{\beta\omega}$, $\boldsymbol{\sigma}_{\beta f}$, $\boldsymbol{\sigma}_{\beta m}$ are the standard deviations of the white noise processes. You can find more details about the derivation from Appendix E.1 of [2].

1.3 Measurement Model

The measurement model for the MEKF can be represented in the standard form

$$
\delta\boldsymbol{z} = \boldsymbol{H}\delta\boldsymbol{x} + \mathtt{V} \tag{37}
$$

where $\delta\boldsymbol{z}$ is the error measurement vector, $\boldsymbol{H}$ is the measurement function, $\delta\boldsymbol{x}$ is the error state vector, and $\mathtt{V}$ is the measurement noise.

In this case, the gyroscope measurement is treated as control input. We have the following two types of measurements:

- Accelerometer: $\tilde{\boldsymbol{a}}^b$
- Magnetometer: $\widetilde{\boldsymbol{m}}^b$

In practice, $\tilde{\boldsymbol{a}}^b$ and $\widetilde{\boldsymbol{m}}^b$ can be acquired directly from the accelerometer and magnetometer, respectively.

Note that Equation 37 in [1] will be used for the calculation of measurement residual for the accelerometer and magnetometer

$$
\boldsymbol{C}_i^b(\boldsymbol{q}) \cong [\boldsymbol{I} - \boldsymbol{\alpha}_\times]\boldsymbol{C}_i^b(\hat{\boldsymbol{q}}) \tag{38}
$$

1.3.1 Accelerometer Measurement Model

The accelerometer measurement model is given by:

$$
\tilde{\boldsymbol{a}}^b = \boldsymbol{C}_i^b(\boldsymbol{q}) \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + \boldsymbol{\beta}_f + \boldsymbol{\eta}_f \tag{39}
$$

Here the acceleration of the rigid body where the IMU is attached to is neglected, and the accelerometer measurement is given by the gravity vector in the inertial frame rotated to the body frame.

Substituting Equation 38 into the above equation, we can get

$$
\begin{aligned}
\tilde{\boldsymbol{a}}^b &= [\boldsymbol{I} - \boldsymbol{\alpha}_\times]\boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\begin{bmatrix}0\\0\\-g\end{bmatrix} + \boldsymbol{\beta}_f + \boldsymbol{\eta}_f \\
&= \boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\begin{bmatrix}0\\0\\-g\end{bmatrix} - \boldsymbol{\alpha}_\times\boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\begin{bmatrix}0\\0\\-g\end{bmatrix} + \boldsymbol{\beta}_f + \boldsymbol{\eta}_f \\
&= \hat{\tilde{\boldsymbol{a}}}^b - \boldsymbol{\alpha} \times \boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\begin{bmatrix}0\\0\\-g\end{bmatrix} + \boldsymbol{\beta}_f + \boldsymbol{\eta}_f \\
&= \hat{\tilde{\boldsymbol{a}}}^b + \boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\begin{bmatrix}0\\0\\-g\end{bmatrix} \times \boldsymbol{\alpha} + \boldsymbol{\beta}_f + \boldsymbol{\eta}_f
\end{aligned}
\tag{40}
$$

$$
\begin{aligned}
\Rightarrow \delta\tilde{\boldsymbol{a}}^b &= \boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\begin{bmatrix}0\\0\\-g\end{bmatrix} \times \boldsymbol{\alpha} + \boldsymbol{\beta}_f + \boldsymbol{\eta}_f \\
&= \begin{bmatrix}\boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\begin{bmatrix}0\\0\\-g\end{bmatrix} \times & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{\alpha}\\ \delta\boldsymbol{v}\\ \delta\boldsymbol{r}\\ \boldsymbol{\beta}_{\boldsymbol{\omega}}\\ \boldsymbol{\beta}_{\boldsymbol{f}}\\ \boldsymbol{\beta}_{\boldsymbol{m}}\end{bmatrix} + \boldsymbol{\eta}_f
\end{aligned}
\tag{41}
$$

1.3.2 Magnetometer Measurement Model

The magnetometer measurement model is given by:

$$
\widetilde{\boldsymbol{m}}^b = \boldsymbol{C}_i^b(\boldsymbol{q})\boldsymbol{m}^i + \boldsymbol{\beta}_m + \boldsymbol{\eta}_m \tag{42}
$$

Similarly, substituting Equation 38, we can get

$$
\begin{aligned}
\widetilde{\boldsymbol{m}}^b &\cong [\boldsymbol{I} - \boldsymbol{\alpha}_\times]\boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\boldsymbol{m}^i + \boldsymbol{\beta}_m + \boldsymbol{\eta}_m \\
&= \boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\boldsymbol{m}^i + \boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\boldsymbol{m}^i \times \boldsymbol{\alpha} + \boldsymbol{\beta}_m + \boldsymbol{\eta}_m \\
&= \hat{\widetilde{\boldsymbol{m}}}^b + \boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\boldsymbol{m}^i \times \boldsymbol{\alpha} + \boldsymbol{\beta}_m + \boldsymbol{\eta}_m
\end{aligned}
\tag{43}
$$

$$
\begin{aligned}
\Rightarrow \delta\widetilde{\boldsymbol{m}}^b &= \boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\boldsymbol{m}^i \times \boldsymbol{\alpha} + \boldsymbol{\beta}_m + \boldsymbol{\eta}_m \\
&= \begin{bmatrix}\boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\boldsymbol{m}^i \times & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{\alpha}\\ \delta\boldsymbol{v}\\ \delta\boldsymbol{r}\\ \boldsymbol{\beta}_{\boldsymbol{\omega}}\\ \boldsymbol{\beta}_{\boldsymbol{f}}\\ \boldsymbol{\beta}_{\boldsymbol{m}}\end{bmatrix} + \boldsymbol{\eta}_m
\end{aligned}
\tag{44}
$$

1.3.3 Full Measurement Model

The full measurement model can be written as

$$
\begin{bmatrix}\delta\tilde{\boldsymbol{a}}^b\\ \delta\widetilde{\boldsymbol{m}}^b\end{bmatrix} = \begin{bmatrix}\begin{bmatrix}\boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\begin{bmatrix}0\\0\\-g\end{bmatrix} \times & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0}\end{bmatrix}\\ \begin{bmatrix}\boldsymbol{C}_i^b(\hat{\boldsymbol{q}})\boldsymbol{m}^i \times & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\end{bmatrix}\delta\boldsymbol{x} + \begin{bmatrix}\boldsymbol{\eta}_f\\ \boldsymbol{\eta}_m\end{bmatrix} \tag{45}
$$

where $\boldsymbol{g}$ and $\boldsymbol{m}^i$ are the gravity vector and the magnetic field vector in the inertial frame, respectively and both are known constants.

We can define the measurement matrix $\boldsymbol{H}$ as

$$\boldsymbol{H} = \begin{bmatrix} \begin{bmatrix} \boldsymbol{C}_i^b(\hat{\boldsymbol{q}}) \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I} & \mathbf{0} \end{bmatrix} \\ \begin{bmatrix} \boldsymbol{C}_i^b(\hat{\boldsymbol{q}}) \boldsymbol{m}^i \times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I} \end{bmatrix} \end{bmatrix} \tag{46}$$

The measurement noise covariance matrix $\boldsymbol{R}_c$ is given by

$$\boldsymbol{R} = \begin{bmatrix} \boldsymbol{\sigma}_a^2 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\sigma}_m^2 \end{bmatrix} \tag{47}$$

where $\boldsymbol{\sigma}_a$ and $\boldsymbol{\sigma}_m$ are the standard deviations of the accelerometer and magnetometer measurements, respectively.

1.4 MEKF Formulation

1.4.1 Model Discretization

We have acquired the error state dynamics and the measurement model as given in Equation 32 and Equation 45.

$$\begin{aligned} \delta\dot{\boldsymbol{x}} &= \boldsymbol{F}\delta\boldsymbol{x} + \boldsymbol{G}\mathtt{w} \\ &= \begin{bmatrix} -\hat{\boldsymbol{\omega}}_\times & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}})\hat{\boldsymbol{f}}_\times^b & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}}) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \delta\boldsymbol{v} \\ \delta\boldsymbol{r} \\ \boldsymbol{\beta}_{\boldsymbol{\omega}} \\ \boldsymbol{\beta}_{\boldsymbol{f}} \\ \boldsymbol{\beta}_{\boldsymbol{m}} \end{bmatrix} + \begin{bmatrix} -\boldsymbol{\eta}_{\boldsymbol{\omega}} \\ -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}})\boldsymbol{\eta}_f \\ \mathbf{0}_{3\times3} \\ \boldsymbol{\nu}_\omega \\ \boldsymbol{\nu}_f \\ \boldsymbol{\nu}_m \end{bmatrix} \end{aligned}$$

The state transition matrix for the discrete system, $\boldsymbol{\Phi}$, is given by:

$$\boldsymbol{\Phi} = e^{\boldsymbol{F}\Delta t} \tag{48}$$

The state transition matrix can be approximated as

$$\begin{aligned} \boldsymbol{\Phi}_{k-1} &= \boldsymbol{I} + \boldsymbol{F}_{k-1}\Delta t \\ &= \boldsymbol{I} + \begin{bmatrix} -\hat{\boldsymbol{\omega}}_{k-1_\times} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}}_{k-1})\hat{\boldsymbol{f}}_{k-1\times}^b & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}}_{k-1}) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \Delta t \end{aligned} \tag{49}$$

Then the state covariance matrix can be updated as

$$\boldsymbol{P}_k^- = \boldsymbol{\Phi}_{k-1}\boldsymbol{P}_{k-1}^+\boldsymbol{\Phi}_{k-1}^T + \boldsymbol{Q}_d \tag{50}$$

where $\boldsymbol{Q}_d$ is the discrete-time process noise covariance matrix and is given by

$$\boldsymbol{Q}_d = \int_0^{\Delta t} e^{\boldsymbol{F}(\boldsymbol{t}-\boldsymbol{\tau})}\boldsymbol{Q}_c e^{\boldsymbol{F}^{\boldsymbol{T}}(\boldsymbol{t}-\boldsymbol{\tau})} d\tau$$

$$= \begin{bmatrix} \Lambda(\boldsymbol{\sigma}_{\boldsymbol{\omega}}^{\mathbf{2}})\Delta t + \Lambda(\boldsymbol{\sigma}_{\boldsymbol{\omega}}^{\mathbf{2}})\Delta\frac{t^3}{3} & \mathbf{0} & \mathbf{0} & -\Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta\omega}}^{\mathbf{2}})\frac{\Delta t^2}{2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Lambda(\boldsymbol{\sigma}_{\boldsymbol{f}}^{\mathbf{2}})\Delta t + \Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta f}}^{\mathbf{2}})\frac{\Delta t^3}{3} & \Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta f}}^{\mathbf{2}})\frac{\Delta t^4}{8} + \Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta f}}^{\mathbf{2}})\Delta\frac{t^2}{2} & \mathbf{0} & -\Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta f}}^{\mathbf{2}})\Delta\frac{t^2}{2} & \mathbf{0} \\ \mathbf{0} & \Lambda(\boldsymbol{\sigma}_{\boldsymbol{f}}^{\mathbf{2}})\frac{\Delta t^2}{2} + \Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta f}}^{\mathbf{2}})\frac{\Delta t^4}{8} & \Lambda(\boldsymbol{\sigma}_{\boldsymbol{f}}^{\mathbf{2}})\frac{\Delta t^3}{3} + \Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta f}}^{\mathbf{2}})\frac{\Delta t^5}{20} & \mathbf{0} & -\Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta f}}^{\mathbf{2}})\frac{\Delta t^3}{6} & \mathbf{0} \\ -\Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta\omega}}^{\mathbf{2}})\frac{\Delta t^2}{2} & \mathbf{0} & \mathbf{0} & \Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta\omega}}^{\mathbf{2}})\frac{\Delta t^2}{2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta f}}^{\mathbf{2}})\Delta\frac{t^2}{2} & -\Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta f}}^{\mathbf{2}})\frac{\Delta t^3}{6} & \mathbf{0} & \Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta f}}^{\mathbf{2}})\Delta t & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \Lambda(\boldsymbol{\sigma}_{\boldsymbol{\beta m}}^{\mathbf{2}})\Delta t \end{bmatrix}$$

Similarly, we have the transition matrix for the measurement model as

$$\boldsymbol{H}_k = \begin{bmatrix} \boldsymbol{C}_i^b(\hat{\boldsymbol{q}}_k^-)\begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I} & \mathbf{0} \\ \boldsymbol{C}_i^b(\hat{\boldsymbol{q}}_k^-)\boldsymbol{m}^i \times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I} \end{bmatrix} \tag{51}$$

1.4.2 MEKF Algorithm

The MEKF can be formulated as follows:

- Initialize the filter with the initial state estimate $\delta\hat{\boldsymbol{x}}_0^+$ and the initial error-state covariance matrix $\boldsymbol{P}_0^+$

$$\delta\hat{\boldsymbol{x}}_0^+ = \mathbf{0} \tag{52}$$

$$\boldsymbol{P}_0^+ = \boldsymbol{E}\left[(\delta\boldsymbol{x}_0 - \delta\hat{\boldsymbol{x}}_0^+)(\delta\boldsymbol{x}_0 - \delta\hat{\boldsymbol{x}}_0^+)^T\right] = \boldsymbol{E}(\delta\boldsymbol{x}_0\delta\boldsymbol{x}_0^T) \tag{53}$$

- For step $k = 1, 2, ...,$ perform the following steps:
  - Update Φ_{k-1} using Equation 49

$$\boldsymbol{\Phi}_{k-1} = \boldsymbol{I} + \boldsymbol{F}_{k-1}\Delta t$$

$$= \boldsymbol{I} + \begin{bmatrix} -\hat{\boldsymbol{\omega}}_{k-1_\times} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}}_{k-1})\hat{\boldsymbol{f}}_{k-1\times}^b & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{C}_b^i(\hat{\boldsymbol{q}}_{k-1}) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix}\Delta t$$

  - Predict the error state and error-state covariance matrix

$$\boldsymbol{P}_k^- = \boldsymbol{\Phi}_{k-1}\boldsymbol{P}_{k-1}^+\boldsymbol{\Phi}_{k-1}^T + \boldsymbol{Q}_d \tag{54}$$

$$\hat{\boldsymbol{x}}_k^- = \boldsymbol{\Phi}_{k-1}\hat{\boldsymbol{x}}_{k-1} \tag{55}$$

  - Update the measurement matrix $\boldsymbol{H}_k$ using Equation 51
  - Compute the Kalman gain $\boldsymbol{K}_k$

$$\boldsymbol{K}_k = \boldsymbol{P}_k^-\boldsymbol{H}_k^T\left(\boldsymbol{H}_k\boldsymbol{P}_k^-\boldsymbol{H}_k^T + \boldsymbol{R}_k\right)^{-1} \tag{56}$$

- Update the error state and error-state covariance matrix

$$\delta\hat{\boldsymbol{x}}_k^+ = \delta\hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k\boldsymbol{H}_k\delta\boldsymbol{x}_k^- \tag{57}$$

$$\boldsymbol{P}_k^+ = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\boldsymbol{P}_k^- \tag{58}$$

- Update the full states

$$\hat{\boldsymbol{q}}_k^+ = \hat{\boldsymbol{q}}_k^- \otimes \begin{bmatrix} 1 \\ \frac{\boldsymbol{\alpha}_k^+}{2} \end{bmatrix} \tag{59}$$

$$\hat{\boldsymbol{r}}_k^+ = \hat{\boldsymbol{r}}_k^- + \delta\boldsymbol{r}_k^+ \tag{60}$$

$$\hat{\boldsymbol{v}}_k^+ = \hat{\boldsymbol{v}}_k^- + \delta\boldsymbol{v}_k^+ \tag{61}$$

2 Math

Inline: Let a, b, and c be the side lengths of right-angled triangle. Then, we know that: $a^2 + b^2 = c^2$

Block without numbering:

$$\sum_{k=1}^{n} k = \frac{n(n+1)}{2}$$

Block with numbering:

As shown in Equation 62.

$$\sum_{k=1}^{n} k = \frac{n(n+1)}{2} \tag{62}$$

More information:
- https://typst.app/docs/reference/math/equation/

3 Citation

You can use citations by using the `#cite` function with the key for the reference and adding a bibliography. Typst supports BibLateX and Hayagriva.

```
#bibliography("bibliography.bib")
```

Single citation [3]. Multiple citations [3], [4]. In text A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin [3]

More information:
- https://typst.app/docs/reference/meta/bibliography/
- https://typst.app/docs/reference/meta/cite/

4 Figures and Tables

header 1	header 2
cell 1	cell 2
cell 3	cell 4

Table 1: Lorem ipsum dolor sit amet.

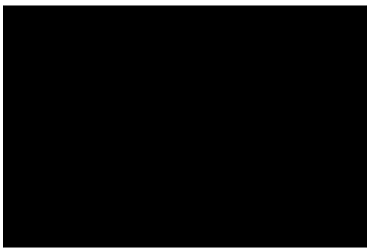

Figure 1: Lorem ipsum dolor sit amet, consectetur adipiscing.

More information

- https://typst.app/docs/reference/meta/figure/

- https://typst.app/docs/reference/layout/table/

5 Referencing

Figure 1 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do., Table 1.

More information:

- https://typst.app/docs/reference/meta/ref/

6 Lists

Unordered list

- Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do.
- Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Numbered list

1. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do.
2. Lorem ipsum dolor sit amet, consectetur adipiscing elit.
3. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor.

More information:

- https://typst.app/docs/reference/layout/enum/
- https://typst.app/docs/reference/meta/cite/

Bibliography

[1] J. M. Maley, "Multiplicative quaternion extended kalman filtering for nonspinning guided projectiles," 2013. [Online]. Available: https://apps.dtic.mil/sti/pdfs/ADA588831.pdf

[2] J. Solà, "Quaternion kinematics for the error-state Kalman filter," Nov. 2017, [Online]. Available: http://arxiv.org/abs/1711.02508

[3] A. Vaswani *et al.*, "Attention is All you Need," in *NIPS*, 2017.

[4] G. E. Hinton, O. Vinyals, and J. Dean, "Distilling the Knowledge in a Neural Network," *ArXiv*, 2015.