Rotation matrices, body vs. earth frame, Euler angles

2023

author: Yerkebulan Massalim

Body frame vs. global frame

Global frame - defines the reference frame
Body frame - defines an object's frame

Problem: obtain the relation between the global and body frame.

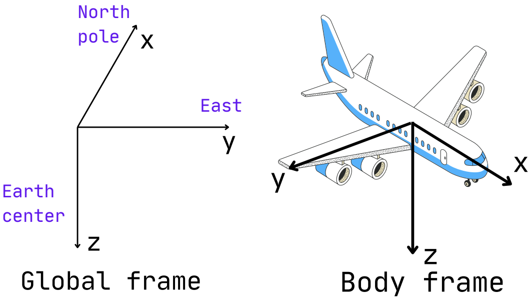


Figure: Global vs. body frame(right handed orthonormal)

Vector rotation

$x_1 = r\cos(\alpha)$ and $y_1 = r\sin(\alpha)$
$x_2 = r\cos(\alpha+\theta) = r(\cos(\alpha)\cos(\theta) - \sin(\alpha)\sin(\theta))$
$y_2 = r\sin(\alpha+\theta) = r(\cos(\alpha)\sin(\theta) + \sin(\alpha)\cos(\theta))$

Using x_1 and y_1 equations,
$x_2 = x_1\cos(\theta) - y_1\sin(\theta)$
$y_2 = x_1\sin(\theta) + y_1\cos(\theta)$

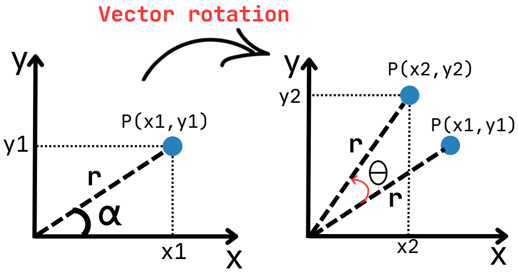


Figure: Vector rotation

Vector rotation, Rotation matrix

$x_2 = x_1 \cos(\theta) - y_1 \sin(\theta) \qquad y_2 = x_1 \sin(\theta) + y_1 \cos(\theta)$

In a matrix form:

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \times \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} \tag{1}$$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = A \times \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} \tag{2}$$

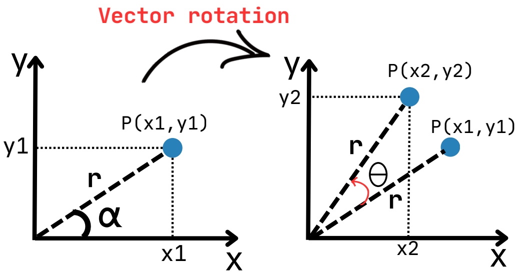


Figure: Vector rotation

Frame rotation

$$x_1 = r\cos(\alpha) \text{ and } y_1 = r\sin(\alpha)$$
$$x_2 = r\cos(\alpha - \theta) = r(\cos(\alpha)\cos(\theta) + \sin(\alpha)\sin(\theta))$$
$$y_2 = r\sin(\alpha - \theta) = r(\cos(\alpha)\sin(\theta) - \sin(\alpha)\cos(\theta))$$

Using x_1 and y_1 equations,

$$x_2 = x_1\cos(\theta) + y_1\sin(\theta)$$
$$y_2 = x_1\sin(\theta) - y_1\cos(\theta)$$

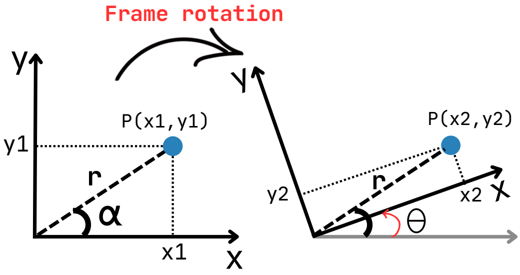


Figure: Frame rotation

Frame rotation, Rotation matrix

$$x_2 = x_1 \cos(\theta) + y_1 \sin(\theta) \qquad y_2 = x_1 \sin(\theta) - y_1 \cos(\theta)$$

In a matrix form:

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \times \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} \tag{3}$$

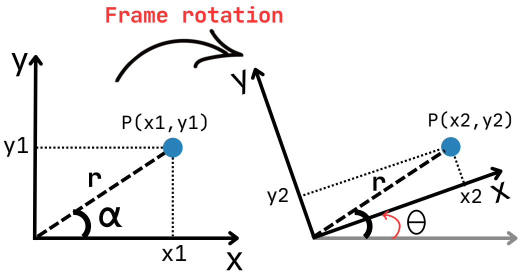


Figure: Frame rotation

Complex numbers: Vector rotation

Vector: $vec_1 = r\cos(\alpha) + \mathbf{i}r\sin(\alpha)$
Rotation: $rot = \cos(\theta) + \mathbf{i}\sin(\theta)$

$vector_1 * rot = (r\cos(\alpha) + \mathbf{i}r\sin(\alpha)) * (\cos(\theta) + \mathbf{i}\sin(\theta)) =$
$r(\cos(\alpha) * \cos(\theta) - \sin(\alpha) * \sin(\theta) + \mathbf{i}(\sin(\alpha) * \cos(\theta) + \cos(\alpha) * \sin(\theta))) =$
$(x_1\cos(\theta) - y_1\sin(\theta)) + \mathbf{i}(x_1\sin(\theta) + y_1\cos(\theta)) \rightarrow$ **Exactly the same result.**

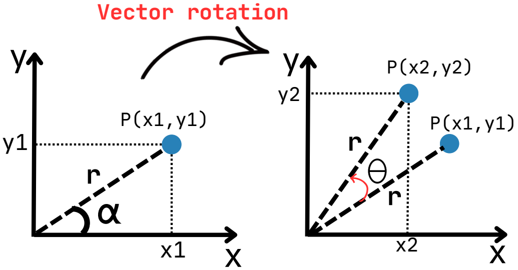


Figure: Vector rotation

Complex numbers: Frame rotation

Vector: $vec_1 = r\cos(\alpha) + \mathbf{i}r\sin(\alpha)$
Rotation: $rot = \cos(\theta) + \mathbf{i}\sin(\theta)$
$\overline{rot} = \cos(\theta) - \mathbf{i}\sin(\theta) \rightarrow$ conjugate
$vector_1 * \overline{rot} = (r\cos(\alpha) + \mathbf{i}r\sin(\alpha)) * (\cos(\theta) - \mathbf{i}\sin(\theta)) =$

$r(\cos(\alpha)*\cos(\theta) + \sin(\alpha)*\sin(\theta) + \mathbf{i}(\sin(\alpha)*\cos(\theta) - \cos(\alpha)*\sin(\theta))) =$
$(x_1\cos(\theta) + y_1\sin(\theta)) + \mathbf{i}(y_1\cos(\theta) - x_1\sin(\theta)) \rightarrow$ **Exactly the same result.**

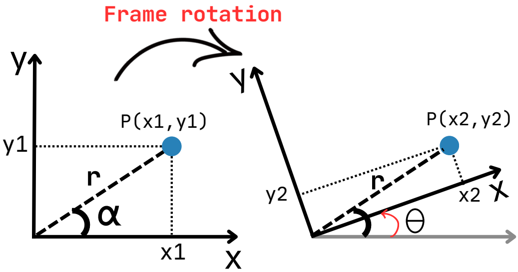


Figure: Frame rotation

Frame rotation and Vector rotation characteristics

$$A_{\theta,frame} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix}$$

$$A_{\theta,vector} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

1. Transpose of vector rotation matrix is equal to frame rotation matrix, or vice versa:
$$A_{\theta,frame} = A^T_{\theta,vector} \tag{4}$$
2. Vector rotation matrix is equal to frame rotation matrix of an opposite angle, or vice versa:
$$A_{\theta,frame} = A^T_{-\theta,vector} \tag{5}$$
3. If frame rotation matrix frame $R_{\theta,frame}$ maps from an old frame to a new frame, $R_{-\theta,frame}$ maps from the new frame to the old frame.
4. Complex numbers:
$$vec\_rot = \cos(\theta) + \mathbf{i}\sin(\theta)$$
$$frame\_rot = \cos(\theta) - \mathbf{i}\sin(\theta)$$
$$vec\_rot = \overline{frame\_rot} \tag{6}$$

3D-space rotation: Euler angles

Any rotation can be described as three successive simple rotations.

ψ(*psi*) - z-axis rotation(yaw)
θ(*theta*) - y-axis rotation (pitch)
ϕ(*phi*) - x-axis rotation (roll)

Variations: $x-y-z, x-z-y, y-z-x, y-x-z, z-x-y,$
$\mathbf{z-y-x}$(**Aircraft EulerAngle Sequence**)$, x-y-x, x-z-x, y-x-y,$
$y-z-y, z-x-z, z-y-z$

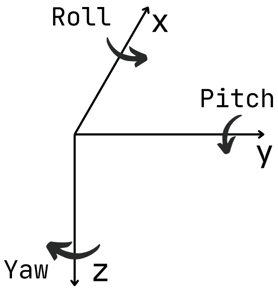


Figure: Roll, pitch, and yaw angles

3D-space rotation: z - y -x example

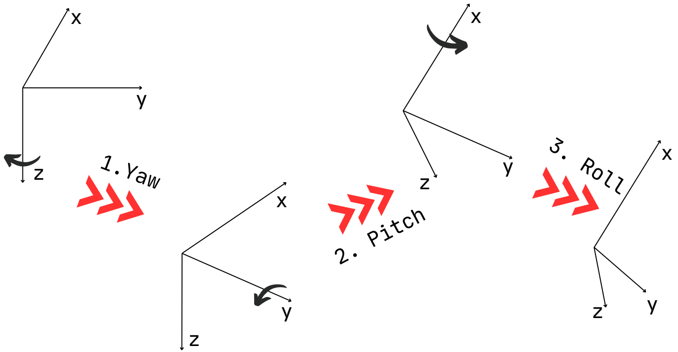


Figure: Roll, pitch, and yaw angles

Rotation matrices for each Yaw-pitch-roll angles

Yaw angle:

$$R_{yaw} = \begin{bmatrix} \cos(\psi) & sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

Pitch angle:

$$R_{pitch} = \begin{bmatrix} \cos(\theta) & 0 & -sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \tag{8}$$

Roll angle:

$$R_{roll} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & sin(\phi) \\ 0 & -\sin(\psi) & \cos(\phi) \end{bmatrix} \tag{9}$$

Full rotation matrix computation

By multiplying roll-pitch-yaw rotation matrices, we can define full rotation matrix. Frame rotation:

$$R_{total} = R_{roll} \times R_{pitch} \times R_{yaw} =$$

$$= \begin{bmatrix} c(\psi)c(\theta) & s(\psi)c(\theta) & -s(\theta) \\ c(\psi)s(\theta)s(\phi) - c(\phi)s(\psi) & c(\phi)c(\psi) + s(\phi)s(\psi)s(\theta) & c(\theta)s(\phi) \\ s(\phi)s(\psi) + c(\phi)c(\psi)s(\theta) & c(\psi)s(\theta)s(\phi) - s(\phi)c(\psi) & c(\theta)c(\phi) \end{bmatrix}$$

$$R_{yaw} = \begin{bmatrix} c(\psi)s(\psi) & 0 & \\ -s(\psi) & c(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad R_{pitch} = \begin{bmatrix} c(\theta) & 0 & -s(\theta) \\ 0 & 1 & 0 \\ s(\theta) & 0 & c(\theta) \end{bmatrix}$$

$$R_{roll} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\phi) & s(\phi) \\ 0 & -s(\psi) & c(\phi) \end{bmatrix}$$

Full rotation matrix

Knowing roll-pitch-yaw angles, we can map vectors (points) from one frame to another.
When defining orientation using Euler angles, we have to identify Euler angles using the IMU data.
Frame rotation:

$$\begin{bmatrix} x_{body} \\ y_{body} \\ z_{body} \end{bmatrix} = R_{total} \begin{bmatrix} x_{earth} \\ y_{earth} \\ z_{earth} \end{bmatrix} \tag{10}$$

Accelerometer, Euler angles 1

Accelerometer measures the acceleration minus gravity in three axes on the body frame.

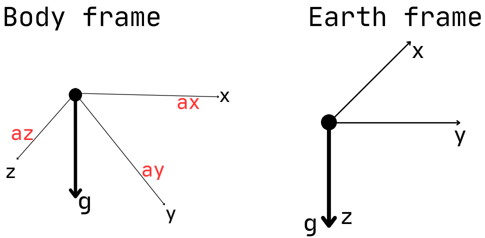
$$\begin{bmatrix} ax_{actual} - gx & ay_{actual} - gy & az_{actual} - gz \end{bmatrix}^T$$

Since we have normalized data, changed the sign in accelerometer readings, and can assume that the object is not accelerating,

$$\begin{bmatrix} gx & gy & gz \end{bmatrix}^T = \begin{bmatrix} ax & ay & az \end{bmatrix}^T$$

$$\begin{bmatrix} ax & ay & az \end{bmatrix}^T = R_{total} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T \tag{11}$$

$$R_{total} = \begin{bmatrix} c(\psi)c(\theta) & s(\psi)c(\theta) & -s(\theta) \\ c(\psi)s(\theta)s(\phi) - c(\phi)s(\psi) & c(\phi)c(\psi) + s(\phi)s(\psi)s(\theta) & c(\theta)s(\phi) \\ s(\phi)s(\psi) + c(\phi)c(\psi)s(\theta) & c(\psi)s(\theta)s(\phi) - s(\phi)c(\psi) & c(\theta)c(\phi) \end{bmatrix}$$



Accelerometer, Euler angles 2

Accelerometer measures the acceleration in three axes on the body frame.

$$ax = -\sin(\theta)$$
$$ay = \cos(\theta)\sin(\phi)$$
$$az = \cos(\theta)\cos(\phi)$$

$$ay^2 + az^2 = cos^2(\theta)(sin^2(\phi) + cos^2(\phi)) = cos(\theta)^2$$

$$cos(\theta) = \pm\sqrt{ay^2 + az^2}$$

Estimated roll and pitch angles, **positive** $\cos\theta$:

$$\phi = \mathbf{atan2}(\mathbf{ay}/\sqrt{\mathbf{ay}^2 + \mathbf{az}^2}, \mathbf{az}/\sqrt{\mathbf{ay}^2 + \mathbf{az}^2}) \quad (12)$$

$$\theta = \mathbf{atan2}(-\mathbf{ax}, \sqrt{\mathbf{ay}^2 + \mathbf{az}^2}) \quad (13)$$

Alternative: **negative** $\cos\theta$:

$$\phi = atan2(-ay/\sqrt{ay^2 + az^2}, -az/\sqrt{ay^2 + az^2}) \quad (14)$$

$$\theta = atan2(-ax, -\sqrt{ay^2 + az^2}) \quad (15)$$

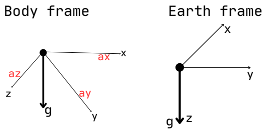


Figure: Gravity in body and earth frame

atan2 function

Problem:

$$atan(\frac{-x1}{-y1}) = atan(\frac{x1}{y1}) = \alpha$$

atan2 function is necessary to consider the sign of $x1$ and $y1$.

$$atan2(-x1, -y1) = \pi + \alpha$$

$$atan2(x1, y1) = \alpha$$

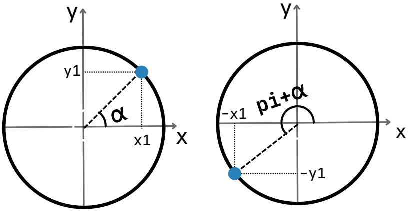


Figure: why we need atan2 function?

Magnetometer, Euler angles 1

Magnetometer measures the magnetic flux in three axes on the body frame.

$$\begin{bmatrix} mx \\ my \\ mz \end{bmatrix} = R_{total} \begin{bmatrix} M_x \\ 0 \\ M_z \end{bmatrix} \tag{16}$$

$$R_{total} = \begin{bmatrix} c(\psi)c(\theta) & s(\psi)c(\theta) & -s(\theta) \\ c(\psi)s(\theta)s(\phi) - c(\phi)s(\psi) & c(\phi)c(\psi) + s(\phi)s(\psi)s(\theta) & c(\theta)s(\phi) \\ s(\phi)s(\psi) + c(\phi)c(\psi)s(\theta) & c(\psi)s(\theta)s(\phi) - s(\phi)c(\psi) & c(\theta)c(\phi) \end{bmatrix}$$

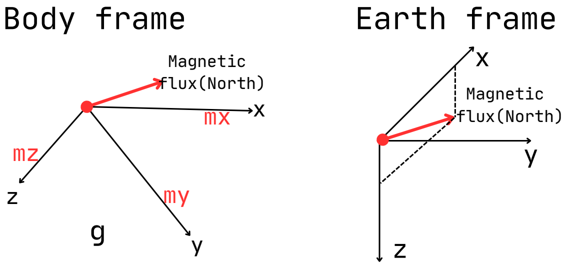


Figure: Magnetic flux in body and earth frame

Magnetometer, Euler angles 2

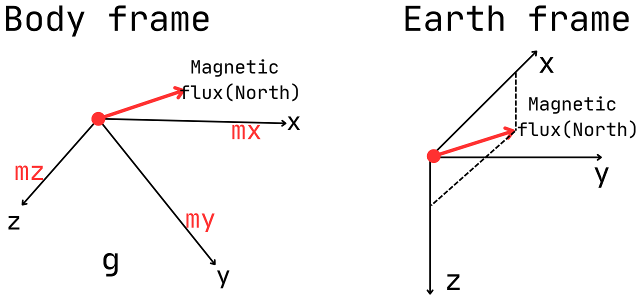
$$mx = c(\psi)c(\theta)M_x - s(\theta)M_z \quad (17)$$
$$my = (c(\psi)s(\theta)s(\phi) - c(\phi)s(\psi))M_x + c(\theta)s(\phi)M_z \quad (18)$$
$$mz = (s(\phi)s(\psi) + c(\phi)c(\psi)s(\theta))M_x + c(\theta)c(\phi)M_z \quad (19)$$

Eq.19$\times s(\phi)$ - Eq.18$\times c(\phi)$:

$$mz \times s(\phi) - my \times c(\phi) = (s(\phi)^2 + c(\phi)^2)s(\psi)Mx = \mathbf{s(\psi)Mx} \quad (20)$$

Eq.17$\times c(\theta)$ + Eq.18$\times s(\phi)s(\theta) + Eq.19 \times c(\phi)s(\theta)$:

$$mx \times c(\theta) + my \times s(\phi)s(\theta) + mz \times c(\phi)s(\theta) = \mathbf{c(\psi)Mx} \quad (21)$$



Magnetometer, Euler angles 3

$$mz \times s(\phi) - my \times c(\phi) = (s(\phi)^2 + c(\phi)^2)s(\psi)Mx = \mathbf{s}(\psi)\mathbf{Mx}$$
$$mx \times c(\theta) + my \times s(\phi)s(\theta) + mz \times c(\phi)s(\theta) = \mathbf{c}(\psi)\mathbf{Mx}$$

Final solution:

$$\psi = atan2(mz \times s(\phi) - my \times c(\phi), \tag{22}$$
$$mx \times c(\theta) + my \times s(\phi)s(\theta) + mz \times c(\phi)s(\theta)) \tag{23}$$

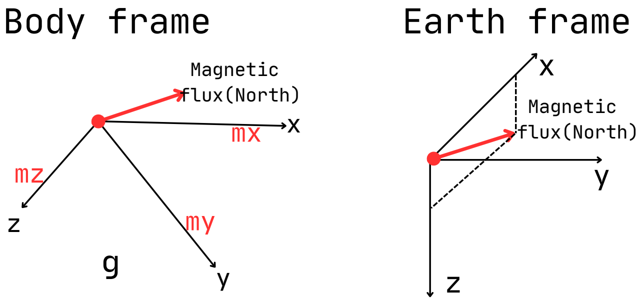


Figure: Magnetic flux in body and earth frame

Gyroscope to Euler angles 1

Gyroscope measures the angular velocity in three axes on the body frame.

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R_{roll} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R_{roll} R_{pitch} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \tag{24}$$

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -s(\theta) \\ 0 & c(\phi) & c(\theta)s(\phi) \\ 0 & -s(\phi) & c(\theta)c(\phi) \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{25}$$

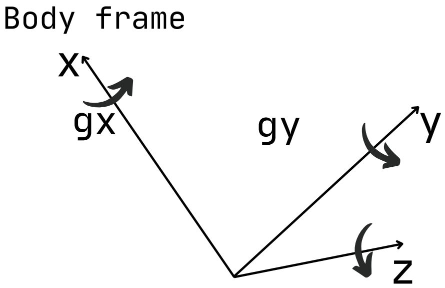


Figure: Gyroscope Illustration

Gyroscope to Euler angles 2

If we inverse equation 25, we get:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & s(\phi)tan(\theta) & c(\phi)tan(\theta) \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi)/c(\theta) & c(\phi)/c(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{26}$$

$$\begin{bmatrix} \phi_{new} \\ \theta_{new} \\ \psi_{new} \end{bmatrix} = \begin{bmatrix} 1 & s(\phi)tan(\theta) & c(\phi)tan(\theta) \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi)/c(\theta) & c(\phi)/c(\theta) \end{bmatrix} \begin{bmatrix} p\Delta t \\ q\Delta t \\ r\Delta t \end{bmatrix} + \begin{bmatrix} \phi_{old} \\ \theta_{old} \\ \psi_{old} \end{bmatrix} \tag{27}$$

$$\begin{bmatrix} p\Delta t & q\Delta t & r\Delta t \end{bmatrix}^T = \begin{bmatrix} gx & gy & gz \end{bmatrix}^T$$

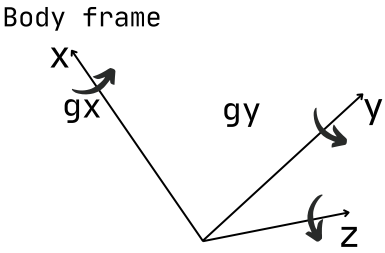


Figure: Gyroscope Illustration

Accelerometer&Magnetometer vs. Gyroscope

Euler angles estimation using the accelerometer & magnetometer is noisy.

Euler angles estimation using the Gyroscope data suffers from drifts.

We have to fuse the estimations to mitigate noise and drifts.

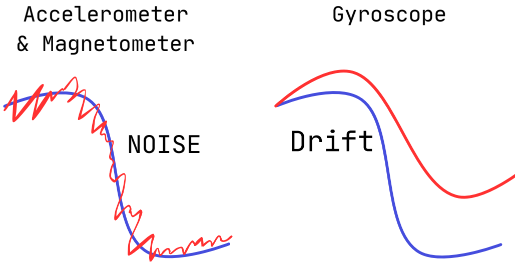


Figure: Noise in accerometer/magnetometer vs. Drifts in Gyroscope

Euler angles, complementary filter

$$\begin{bmatrix} \phi_{comp.} \\ \theta_{comp.} \\ \psi_{comp.} \end{bmatrix} = ALPHA \times \begin{bmatrix} \phi_{gyro} \\ \theta_{gyro} \\ \psi_{gyro} \end{bmatrix} + (1 - ALPHA) \times \begin{bmatrix} \phi_{acc\_mag} \\ \theta_{acc\_mag} \\ \psi_{acc\_mag} \end{bmatrix} \tag{28}$$

Rewriting the equation:

$$\begin{bmatrix} \phi_{comp.} \\ \theta_{comp.} \\ \psi_{comp.} \end{bmatrix} = \begin{bmatrix} \phi_{gyro} \\ \theta_{gyro} \\ \psi_{gyro} \end{bmatrix} + (1 - ALPHA) \times \begin{bmatrix} \phi_{acc\_mag} - \phi_{gyro} \\ \theta_{acc\_mag} - \theta_{gyro} \\ \psi_{acc\_mag} - \psi_{gyro} \end{bmatrix}$$

Steps to be taken:

1. Receive IMU data
2. Remove biases / Normalize/ Scale IMU data
3. Compute Euler angles using accelereometer/magnetometer
4. Compute new Euler angles using the gyroscope readings
5. Fuse acc/mag and gyroscope-computed Euler angles using the complementary filter

Gimbal lock or why we do not use Euler angles

$X - Y - Z$ rotation $\theta = 90\,\text{deg}$

$$R_{total} = \begin{bmatrix} 0 & 0 & -1 \\ c(\psi)s(\phi) - c(\phi)s(\psi) & c(\phi)c(\psi) + s(\phi)s(\psi) & 0 \\ s(\psi)s(\phi) + c(\phi)c(\psi) & c(\psi)s(\phi) - s(\phi)c(\psi) & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & -1 \\ s(\psi - \phi) & c(\phi - \psi) & 0 \\ c(\psi - \phi) & s(\psi - \phi) & 0 \end{bmatrix}$$

We can identify $\psi - \phi$, not individual angles.

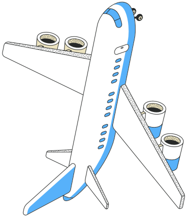


Figure: Gimbal Illustration

QUATERNIONs

$q = s + x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k}, \qquad s, x, y, z \in \mathbb{R}$
$\boldsymbol{i}^2 = \boldsymbol{j}^2 = \boldsymbol{k}^2 = -1,$
$\boldsymbol{ij} = \boldsymbol{k}, \boldsymbol{jk} = \boldsymbol{i}, \boldsymbol{ki} = \boldsymbol{j}, \boldsymbol{ji} = -\boldsymbol{k}, \boldsymbol{kj} = -\boldsymbol{i}, \boldsymbol{ik} = -\boldsymbol{j}$
Common expression:
$q = [s, \boldsymbol{v}], \qquad s \in \mathbb{R}, \boldsymbol{v} \in \mathbb{R}^3$

Algebraic definitions

$q_a = [s_a, \boldsymbol{a}] \qquad q_b = [s_b, \boldsymbol{b}]$
Adding and subtracting: $q_a + q_b = [s_a \pm s_b, \boldsymbol{a} \pm \boldsymbol{b}]$
Multiplying a quaternion by a scalar: $\lambda q_a = [\lambda s_a, \lambda \boldsymbol{a}]$
Pure quaternion: $q = [0, \boldsymbol{v}]$

Norm of a quaternion: $q = [s, x, y, z]$,
$|q| = \sqrt{s^2 + x^2 + y^2 + z^2}$

Unit norm: $|q| = 1$
Conjugate: $q^* = [s, -\boldsymbol{v}]$
Inverse: $q^{-1} = \frac{q^*}{|q|^2}$

Quaternion Products

$q_a = [s_a, \boldsymbol{a}] \qquad q_b = [s_b, \boldsymbol{b}]$
$q_a q_b = (s_a + x_a\boldsymbol{i} + y_a\boldsymbol{j} + z_a\boldsymbol{k})(s_b + x_b\boldsymbol{i} + y_b\boldsymbol{j} + z_b\boldsymbol{k}) =$
$(s_a s_b - x_a x_b - y_a y_b - z_a z_b) + (s_a x_b + x_a s_b + y_a z_b - z_a y_b)\boldsymbol{i} +$
$(s_a y_b - x_a z_b + y_a s_b + z_a x_b)\boldsymbol{j} + (s_a z_b + x_a y_b - y_a x_b + z_a s_b)\boldsymbol{k}$
Matrix form

$$q_a q_b = \begin{bmatrix} s_a & -x_a & -y_a & -z_a \\ x_a & s_a & -z_a & y_a \\ y_a & z_a & s_a & -x_a \\ z_a & -y_a & x_a & s_a \end{bmatrix} \begin{bmatrix} s_b \\ x_b \\ y_b \\ z_b \end{bmatrix} = \boldsymbol{L}(q_a) q_b =$$

$$\begin{bmatrix} s_b & -x_b & -y_b & -z_b \\ x_b & s_b & z_b & -y_b \\ y_b & -z_b & s_b & x_b \\ z_b & y_b & -x_b & s_b \end{bmatrix} \begin{bmatrix} s_a \\ x_a \\ y_a \\ z_a \end{bmatrix} = \boldsymbol{R}(q_b) q_a$$

Quaternion rotation

Vector rotation: Rotate a point $\boldsymbol{p}(x_p, y_p, z_p)$ around vector $\hat{\boldsymbol{v}}(x, y, z)$ by θ angle.
Solution: define pure quaternion $p = [0, x_p, y_p, z_p]$ and rotation quaternion $q = [\cos(\frac{\theta}{2}), \sin(\frac{\theta}{2})x, \sin(\frac{\theta}{2})y, \sin(\frac{\theta}{2})z]$
Do the multiplication:

$$\boldsymbol{p' = qpq^{-1}}$$

Frame rotation: Rotate a frame around vector $\boldsymbol{\hat{v}(x, y, z)}$ by $\boldsymbol{\theta}$ angle. Find the coordinate of point $\boldsymbol{p(x_p, y_p, z_p)}$ with respect to new vector frame
Solution: define pure quaternion $\boldsymbol{p = [0, x_p, y_p, z_p]}$ and rotation quaternion $\boldsymbol{q = [\cos(\frac{\theta}{2}), \sin(\frac{\theta}{2})x, \sin(\frac{\theta}{2})y, \sin(\frac{\theta}{2})z]}$
Do the multiplication:

$$\boldsymbol{p' = q^{-1}pq}$$

Quaternion rotation example

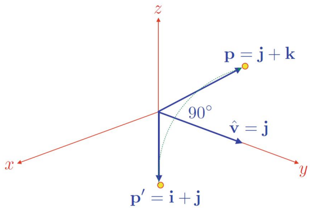


Figure: The point(0,1,1) is rotated around y-axis to 90 degrees

$$q = [\cos(45), 0, \sin(45), 0], p = [0, 0, 1, 1]$$
$$qpq^{-1} = (qp)q^{-1} = \boldsymbol{L}(\boldsymbol{q})pq^{-}1 = aq^{-}1 = \boldsymbol{R}(\boldsymbol{q}^{-1})\boldsymbol{a} = \boldsymbol{R}(\boldsymbol{q}^{-1})\boldsymbol{a}$$

$$\boldsymbol{a} = \begin{bmatrix} \cos(45) & 0 & -\sin(45) & 0 \\ 0 & \cos(45) & 0 & \sin(45) \\ \sin(45) & 0 & \cos(45) & 0 \\ 0 & -\sin(45) & 0 & \cos(45) \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -\sin(45) \\ \sin(45) \\ \cos(45) \\ \cos(45) \end{bmatrix}$$

Quaternion rotation example(cont.)

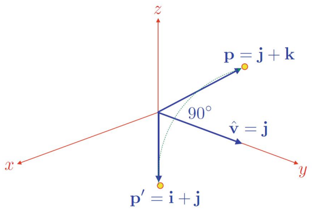


Figure: The point(0,1,1) is rotated around y-axis to 90 degrees

$$\boldsymbol{Ra} = \begin{bmatrix} \cos(45) & 0 & \sin(45) & 0 \\ 0 & \cos(45) & 0 & \sin(45) \\ -\sin(45) & 0 & \cos(45) & 0 \\ 0 & -\sin(45) & 0 & \cos(45) \end{bmatrix} \begin{bmatrix} -\sin(45) \\ \sin(45) \\ \cos(45) \\ \cos(45) \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

Quaternion to rotation matrix

$$R(q) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (29)$$

Vector rotation: $p' = qpq^{-1}$

$$\begin{bmatrix} x_{new} \\ y_{new} \\ z_{new} \end{bmatrix} = R(q) \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (30)$$

Frame rotation: $p' = q^{-1}pq$

$$\begin{bmatrix} x_{new} \\ y_{new} \\ z_{new} \end{bmatrix} = R^T(q) \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (31)$$

Multiple rotations using quaternions

Sequential Vector rotations
First rotation q_1: $q_1 p q_1^{-1}$
Second rotation q_2: $q_2(q_1 p q_1^{-1})q_2^{-1} = (q_2 q_1)p(q_1^{-1}q_2^{-1})$
Since

$$q_1^{-1}q_2^{-1} = (q_2 q_1)^{-1}, \tag{32}$$

$(q_2 q_1)p(q_1^{-1}q_2^{-1}) = (q_2 q_1)p(q_2 q_1)^{-1}$
Key point: n sequential rotations of a **vector** can be defined as $q_n q_{n-1} \ldots q_1$

Sequential Frame rotations
First rotation q_1: $q_1^{-1} p q_1$
Second rotation q_2: $q_2^{-1}(q_1^{-1} p q_1)q_2 = (q_2^{-1}q_1^{-1})p(q_1 q_2)$
Since

$$q_2^{-1}q_1^{-1} = (q_1 q_2)^{-1}, \tag{33}$$

$(q_2^{-1}q_1^{-1})p(q_1 q_2) = (q_1 q_2)^{-1}p(q_1 q_2)$
Key point: n sequential rotations of a **frame** can be defined as $q_1 q_2 \ldots q_n$

Accelerometer to Quaternion

$$R^T(q) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 + q_0 q_3) & 2(q_1 q_3 - q_0 q_2) \\ 2(q_1 q_2 - q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 + q_0 q_1) \\ 2(q_1 q_3 + q_0 q_2) & 2(q_2 q_3 - q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{34}$$

Frame rotation: $p' = q^{-1} p q$

$$\begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix} = R^T(q) \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{35}$$

$$a_x = 2(q_1 q_3 - q_0 q_2) \tag{36}$$

$$a_y = 2(q_2 q_3 + q_0 q_1) \tag{37}$$

$$a_z = q_0^2 - q_1^2 - q_2^2 + q_3^2 \tag{38}$$

Accelerometer to Quaternion

We assume that $q_3 = 0$. Then,
$a_x = -2q_0q_2$, $a_y = 2q_0q_1$, and $a_z = q_0^2 - q_1^2 - q_2^2$

$$a_z + 1 = q_0^2 - q_1^2 - q_2^2 + q_0^2 + q_1^2 + q_2^2 = 2q_0^2 \tag{39}$$

Let's denote $\lambda_1 = \sqrt{\frac{a_z + 1}{2}}$. Using this expression, we can compute q_acc as follows

$$q_{acc} = \begin{bmatrix} \lambda_1 & \frac{a_y}{2\lambda_1} & -\frac{a_x}{2\lambda_1} & 0 \end{bmatrix}^T \tag{40}$$

To avoid singularity, we define another solution:
$q_2 = 0$, $a_x = 2q_1q_3$, $a_y = 2q_0q_1$, and $a_z = q_0^2 - q_1^2 + q_3^2$
$\lambda_2 = \sqrt{\frac{1 - a_z}{2}}$

$$q_{acc} = \begin{bmatrix} \frac{a_y}{2\lambda_2} & \lambda_2 & 0 & \frac{a_x}{2\lambda_2} \end{bmatrix}^T \tag{41}$$

Accelerometer to Quaternion

Final solution:

$a_z > 0,\ \lambda_1 = \sqrt{\frac{a_z + 1}{2}}$

$$q_{acc} = \begin{bmatrix} \lambda_1 & \frac{a_y}{2\lambda_1} & -\frac{a_x}{2\lambda_1} & 0 \end{bmatrix}^T \tag{42}$$

$a_z < 0,\ \lambda_2 = \sqrt{\frac{1 - a_z}{2}}$

$$q_{acc} = \begin{bmatrix} \frac{a_y}{2\lambda_2} & \lambda_2 & 0 & \frac{a_x}{2\lambda_2} \end{bmatrix}^T \tag{43}$$

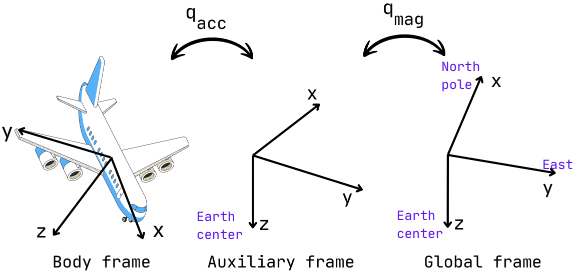


Figure: auxiliary frame

Accelerometer to Quaternion

Final solution:

$a_z > 0, \ \lambda_1 = \sqrt{\frac{a_z + 1}{2}}$

$$q_{acc} = \begin{bmatrix} \lambda_1 & \frac{a_y}{2\lambda_1} & -\frac{a_x}{2\lambda_1} & 0 \end{bmatrix}^T \tag{44}$$

$a_z < 0, \ \lambda_2 = \sqrt{\frac{1 - a_z}{2}}$

$$q_{acc} = \begin{bmatrix} \frac{a_y}{2\lambda_2} & \lambda_2 & 0 & \frac{a_x}{2\lambda_2} \end{bmatrix}^T \tag{45}$$

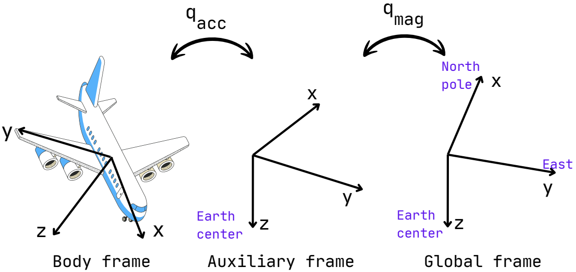


Figure: auxiliary frame

Magnetometer to Quaternion

Let's compute the magnetometer readings in auxiliary frame. Since, q_acc defines a rotation from auxiliary to body frame, q_{acc}^{-1} defines a rotation from body to auxiliary frame. In that case, $R((q_{acc}^{-1})^{-1}) = R(q_{acc})$ will map body frame magnetometer readings to auxiliary frame magnetometer readings.

$$\begin{bmatrix} l_x \\ l_y \\ M_z \end{bmatrix} = R(q_{acc}) \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} \tag{46}$$

Magnetometer to Quaternion

Let's define a quaternion that rotates along the z-axis:

$$q_{mag} = \begin{bmatrix} q_{0,mag} & 0 & 0 & q_{3,mag} \end{bmatrix}^T \tag{47}$$

Based on this quaternion, one can define a rotation matrix:

$$R^T = \begin{bmatrix} q_{0,mag}^2 - q_{3,mag}^2 & 0 & 0 \\ -2q_{0,mag}q_{3,mag} & q_{0,mag}^2 - q_{3,mag}^2 & 0 \\ 0 & 0 & q_0^2 + q_3^2 \end{bmatrix} \tag{48}$$

If we do rotation:

$$\begin{bmatrix} l_x \\ l_y \\ M_z \end{bmatrix} = R^T \begin{bmatrix} M_x \\ 0 \\ M_z \end{bmatrix} \tag{49}$$

Finally, we obtain the following equations:

$$l_x = (q_{0,mag}^2 - q_{3,mag}^2)M_x \tag{50}$$

$$l_y = -2q_{0,mag}q_{3,mag}M_x \tag{51}$$

$$M_z = (q_0^2 + q_3^2)M_z \tag{52}$$

Magnetometer to Quaternion

$$l_x = (q_{0,mag}^2 - q_{3,mag}^2)M_x$$
$$l_y = -2q_{0,mag}q_{3,mag}M_x$$
$$l_z = (q_0^2 + q_3^2)l_z$$

If we solve these equations,

$l_x > 0$

$$q_{mag} = \begin{bmatrix} \frac{\sqrt{M_x^2 + l_x M_x}}{\sqrt{2}M_x} & 0 & 0 & -\frac{l_y}{\sqrt{2}\sqrt{M_x^2 + l_x M_x}} \end{bmatrix} \tag{53}$$

$l_x < 0$

$$q_{mag} = \begin{bmatrix} -\frac{l_y}{\sqrt{2}\sqrt{M_x^2 - l_x M_x}} & 0 & 0 & \frac{\sqrt{M_x^2 - l_x M_x}}{\sqrt{2}M_x} \end{bmatrix} \tag{54}$$

Magnetometer/Accelerometer to Quaternion

Accelerometer:

$$q_{acc} = \begin{bmatrix} \lambda_1 & \frac{a_y}{2\lambda_1} & -\frac{a_x}{2\lambda_1} & 0 \end{bmatrix}^T, a_z > 0, \lambda_1 = \sqrt{\frac{a_z + 1}{2}}$$

$$q_{acc} = \begin{bmatrix} \frac{a_y}{2\lambda_2} & \lambda_2 & 0 & \frac{a_x}{2\lambda_2} \end{bmatrix}^T, a_z < 0, \lambda_2 = \sqrt{\frac{1 - a_z}{2}}$$

Magnetometer:

$$q_{mag} = \begin{bmatrix} \frac{\sqrt{M_x^2 + l_x M_x}}{\sqrt{2} M_x} & 0 & 0 & -\frac{l_y}{\sqrt{2}\sqrt{M_x^2 + l_x M_x}} \end{bmatrix}, l_x > 0$$

$$q_{mag} = \begin{bmatrix} \frac{l_y}{-\sqrt{2}\sqrt{M_x^2 - l_x M_x}} & 0 & 0 & \frac{\sqrt{M_x^2 - l_x M_x}}{\sqrt{2} M_x} \end{bmatrix}, l_x < 0$$

Final solution:

$$q_{final} = q_{mag} q_{acc} \tag{55}$$

Update Quaternions using the gyroscope readings

Gyroscope readings: $[\omega_x, \omega_y, \omega_z]$.

Converting to quaternions:
$\omega = \sqrt{\omega_x^2 + \omega_y^2 + \omega_z^2}, x_g = \omega_x/\omega, y_g = \omega_y/\omega, z_g = \omega_z/\omega$
$q_{gyro} = [\cos(\omega/2), x_g \sin(\omega/2), y_g \sin(\omega/2), z_g \sin(\omega/2)]^T$

Since $\omega \approx 0, \cos(\omega/2) = 1, \sin(\omega/2) = \omega/2,$
$q_{gyro} = [1, x_g\omega/2, y_g\omega/2, z_g\omega/2] = [1, \omega_x/2, \omega_y/2, \omega_z/2]^T$

$q_{new} = q_{old} q_{gyro} = R(q_{gyro}) q_{old}$

$$q_{new} = \begin{bmatrix} 1 & -\omega_x/2 & -\omega_y/2 & -\omega_z/2 \\ \omega_x/2 & 1 & \omega_z/2 & -\omega_y/2 \\ -\omega_y/2 & -\omega_z/2 & 1 & \omega_x/2 \\ \omega_z/2 & \omega_y/2 & -\omega_x/2 & 1 \end{bmatrix} q_{old}$$

Complementary filter based on quaternions

Complementary Filter:

$$q_{comp.} = ALPHA \times q_{gyro} + (1 - ALPHA) \times q_{acc\_mag}$$

Steps to be taken:

1. Compute the quaternions using the accelerometer and magnetometer.
2. Update $q_{compl.old}$ using the gyroscope.
3. Change the sign of q_{acc_mag} if the dot product of the quaternions is negative.
4. Apply the Complementary Filter.
5. Normalize q_{compl}.

Linear Kalman Filter, summary

Prediction, phase 1:

$$x_{n,pred} = F\hat{x}_{n-1} + Gu_{n-1} \tag{56}$$

$$\Sigma_{x_{n,pred}} = F\Sigma_{\hat{x}_{n-1}}F^T + Q \tag{57}$$

Correction, phase 2 ($y_{meas} = Cx$):

$$\mathbf{k_g} = \frac{\mathbf{\Sigma_{n_{n,pred}} C^T}}{\mathbf{\Sigma_{x_{n,meas.}} + C\Sigma_{x_{n,pred}} C^T}} \tag{58}$$

$$\mathbf{\hat{x}_n} = \mathbf{x_{n,pred}} + \mathbf{k_g}(\mathbf{y_{meas}} - \mathbf{Cx_{n,pred}}) \tag{59}$$

$$\mathbf{\Sigma_{\hat{x}_n}} = \mathbf{\Sigma_{x_{n,pred}}} - \mathbf{k_g C \Sigma_{x_{n,pred}}} \tag{60}$$

Jacobian Matrix

Example: Jacobian matrix

$$y_1 = \sin(x_1) + {x_2}^2$$

$$y_2 = \tan(x_1) + {x_2}^3$$

Jacobian Matrix:

$$\mathbf{J}_y = \frac{\partial y}{\partial x} = \begin{bmatrix} cos(x_1) & 2x_2 \\ \dfrac{1}{1 + cos(x_1)^2} & 3{x_2}^2 \end{bmatrix}$$

If the covariance matrix of $\mathbf{x}$ is

$$\Sigma_x = \begin{bmatrix} 2 & 3 \\ 0 & 2 \end{bmatrix}$$

Then, the covariance matrix of $\mathbf{y}$

$$\Sigma_y = \mathbf{J}_y \Sigma_x \mathbf{J}_y^T$$

Extended Kalman Filter, summary

Prediction, phase 1:

$$\boldsymbol{x_{n,pred}} = \boldsymbol{f}(\boldsymbol{\hat{x}_{n-1}}) + \boldsymbol{Gu_{n-1}} \tag{61}$$

$$\boldsymbol{\Sigma_{x_{n,pred}}} = \mathbf{J_f} \boldsymbol{\Sigma_{\hat{x}_{n-1}}} \mathbf{J_f}^{\boldsymbol{T}} + \boldsymbol{Q}, \tag{62}$$

where $\mathbf{J}_f = \dfrac{\partial f}{\partial \hat{x}_{n-1}}$

Correction, phase 2 ($y_{meas} = h(x)$):

$$\mathbf{k_g} = \frac{\mathbf{\Sigma_{n_{n,pred}} J_h}^{\mathbf{T}}}{\mathbf{\Sigma_{x_{n,meas.}}} + \mathbf{J_h \Sigma_{x_{n,pred}} J_h}^{\mathbf{T}}} \tag{63}$$

$$\mathbf{\hat{x}_n} = \mathbf{x_{n,pred}} + \mathbf{k_g}(\mathbf{y_{meas}} - \mathbf{h}(\mathbf{x_{n,pred}})) \tag{64}$$

$$\mathbf{\Sigma_{\hat{x}_n}} = \mathbf{\Sigma_{x_{n,pred}}} - \mathbf{k_g J_h \Sigma_{x_{n,pred}}}, \tag{65}$$

where $\mathbf{J}_h = \dfrac{\partial h}{\partial \hat{x}_{n,pred}}$

Attitude estimation, Extended Kalman Filter, Prediction Phase

Gyroscope readings: $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T$.
State: $q = [q_0, q_1, q_2, q_3]^T$

Prediction, phase 1:

$$\boldsymbol{q_{n,pred}} = \boldsymbol{q_{n-1} q_{gyro}} = \boldsymbol{R(q_{gyro}) q_{n-1}} \tag{66}$$

where F is

$$R(q_{gyro}) = \begin{bmatrix} 1 & -\omega_x/2 & -\omega_y/2 & -\omega_z/2 \\ \omega_x/2 & 1 & \omega_z/2 & -\omega_y/2 \\ -\omega_y/2 & -\omega_z/2 & 1 & \omega_x/2 \\ \omega_z/2 & \omega_y/2 & -\omega_x/2 & 1 \end{bmatrix}$$

Covariance prediction:

$$\Sigma_{q_{n,pred}} = R \Sigma_{\hat{q}_{n-1}} R^T + M \boldsymbol{\omega} M^T,$$

where M is

$$M = \begin{bmatrix} -q_{1,n-1} & -q_{2,n-1} & -q_{3,n-1} \\ q_{0,n-1} & -q_{3,n-1} & q_{2,n-1} \\ q_{3,n-1} & q_{0,n-1} & -q_{1,n-1} \\ -q_{2,n-1} & q_{1,n-1} & -q_{0,n-1} \end{bmatrix}$$

Attitude estimation, Extended Kalman Filter, Prediction Phase

Gyroscope readings: $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T$.

Prediction, phase 1:

$$\boldsymbol{q}_{n,pred} = \boldsymbol{q}_{n-1}\boldsymbol{q}_{gyro} = \boldsymbol{R}\boldsymbol{q}_{n-1} \tag{67}$$

Covariance prediction:

$$\Sigma_{q_{n,pred}} = R\Sigma_{\hat{q}_{n-1}}R^T + M\boldsymbol{\omega}M^T,$$

where M is
Measurement Prediction:

$$\begin{bmatrix} ax_p \\ ay_p \\ az_p \\ mx_p \\ my_p \\ mz_p \end{bmatrix} = \begin{bmatrix} 2q_{1,p}q_{3,p} - 2q_{0,p}q_{2,p} \\ 2q_{2,p}q_{3,p} + 2q_{0,p}q_{1,p} \\ q_{0,p}^2 - q_{1,p}^2 - q_{2,p}^2 + q_{3,p}^2 \\ (q_{0,p}^2 + q_{1,p}^2 - q_{2,p}^2 - q_{3,p}^2)M_x + (2q_{1,p}q_{3,p} - 2q_{0,p}q_{2,p})M_z \\ (2q_{1,p}q_{2,p} - 2q_{0,p}q_{3,p})M_x + (2q_{2,p}q_{3,p} + 2q_{0,p}q_{1,p})M_z \\ (2q_{1,p}q_{3,p} + 2q_{0,p}q_{2,p})M_x + (q_{0,p}^2 - q_{1,p}^2 - q_{2,p}^2 + q_{3,p}^2)M_z \end{bmatrix} \tag{68}$$

Attitude estimation, Extended Kalman Filter, Correction Phase

Jacobian Matrix:

$$J_h = \begin{bmatrix} -2q_2 & 2q_3 & -2q_0 & 2q_1 \\ 2q_1 & 2q_0 & 2q_{3,p} & q_3 \\ 2q_0 & -2q_1 & -2q_2 & 2q_3 \\ 2q_0 M_x - 2q_2 M_z & 2q_1 M_x + 2q_3 M_z & -2q_2 M_x - 2q_0 M_z & -2q_3 M_x + 2q_1 M_z \\ 2q_1 M_z & 2q_0 M_z & 2q_3 M_z & q_3 M_z \\ 2q_0 M_z & -2q_1 M_z & -2q_2 M_z & 2q_3 M_z \end{bmatrix} \tag{69}$$

Correction, phase 2:

$$\mathbf{K_g} = \frac{\mathbf{\Sigma_{q_{pred}}} \mathbf{J_h}^{\mathbf{T}}}{\mathbf{\Sigma_{q_{meas}}} + \mathbf{J_h} \mathbf{\Sigma_{q_{pred}}} \mathbf{J_h}^{\mathbf{T}}} \tag{70}$$

$$\begin{bmatrix} \hat{q}_0 \\ \hat{q}_1 \\ \hat{q}_2 \\ \hat{q}_3 \end{bmatrix} = \begin{bmatrix} q_{0,pred} \\ q_{1,pred} \\ q_{2,pred} \\ q_{3,pred} \end{bmatrix} + \mathbf{K_g}(\begin{bmatrix} ax_{meas} \\ ay_{meas} \\ az_{meas} \\ mx_{meas} \\ my_{meas} \\ mz_{meas} \end{bmatrix} - \begin{bmatrix} ax_p \\ ay_p \\ az_p \\ mx_p \\ my_p \\ mz_p \end{bmatrix}) \tag{71}$$

$$\mathbf{\Sigma_{\hat{q}}} = \mathbf{\Sigma_{q_{pred}}} - \mathbf{K_g} \mathbf{J_h} \mathbf{\Sigma_{q_{pred}}}, \tag{72}$$