ME 467 – ECE 564: Robotics and Automated Systems
Spring 2025 — Project II: Localization

In this project, we will implement a couple (orientation) localization algorithms and compare them to a third algorithm, the so-called Mahony filter [1]. Mahony filter, which was first introduced in the seminal paper [4], is a type of nonlinear complementary filter. In our implementation, the orientation of our cell phone $\{C\}$ with respect to the inertial frame $\{0\}$ will be represented by the rotation matrix $\boldsymbol{R}$ or the associated quaternion $\boldsymbol{q}$.

The purpose of these algorithms is to obtain an estimate of the orientation with an inertial measurement unit (IMU) strapped on a rigid body. Today, every cell phone has an IMU incorporated to its circuitry. For Android and iOS users, we will make use of HyperIMU [3] and SensorLog [5], respectively, in order to collect data from the on-board IMU.

Question 1 .... *20 (main) + 0 (bonus) points*

Construct an orientation simulator over the quaternions in your favorite programming language. This simulator should take the (body) angular velocity $\boldsymbol{\omega}$ and an interval δt as inputs and produce the resulting quaternion $\boldsymbol{q}(t+\delta t)$ at time $t+\delta t$ given the current quaternion $\boldsymbol{q}(t)$ at time t.

We achieve this by numerical integration. The orientation kinematics is given by the differential equation

$$\dot{\boldsymbol{q}} = \frac{1}{2}\boldsymbol{q} \circ \boldsymbol{p}(\boldsymbol{\omega}) = \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega})\boldsymbol{q}, \tag{1}$$

where the map $\boldsymbol{p}$ takes a three-vector $\boldsymbol{\omega}$ and outputs a pure quaternion: $\boldsymbol{\omega} \overset{\boldsymbol{p}}{\mapsto} 0\langle\boldsymbol{\omega}\rangle$. The operation $\circ$ represents quaternion multiplication. The last equality expresses the quaternion multiplication as a matrix multiplication of the quaternion $\boldsymbol{q}$, viewed as a 4-vector, with the 4-by-4 matrix $\boldsymbol{\Omega}(\boldsymbol{\omega})$, defined by

$$\boldsymbol{\Omega}(\boldsymbol{\omega}) \triangleq \begin{bmatrix} 0 & -\boldsymbol{\omega}^\top \\ \boldsymbol{\omega} & -\boldsymbol{\omega}_\times \end{bmatrix},$$

with $\boldsymbol{\omega}_\times$ representing the skew-symmetric form of $\boldsymbol{\omega}$. Note that $\boldsymbol{\Omega}(\boldsymbol{\omega})^2 = -\|\boldsymbol{\omega}\|^2\boldsymbol{I}$.

How does one integrate the orientation kinematics (1)? First, we assume that the integration period is very short: $\delta t \ll {}^1\!/_f$, where f is the frequency of the fastest motion the rigid-body can undergo. This allows us to assume that $\boldsymbol{\omega}$ is constant during an interval of operation $[t_0, t_0+\delta t)$. We then solve the differential equation (1) in closed-form under this assumption to obtain

$$\boldsymbol{q}(t) = e^{1/2\boldsymbol{\Omega}(\boldsymbol{\omega})t}\boldsymbol{q}(t_0), \quad t \in [t_0, t_0+\delta t).$$

If we express this equation in terms of the integration step k so that $t = k\delta t$, we obtain the following update equation

$$\boldsymbol{q}_{k+1} = e^{1/2\boldsymbol{\Omega}(\boldsymbol{\omega})\delta t}\boldsymbol{q}_k.$$

Finally, note that the exponential in this formula may be evaluated in closed-form using the formula [6]

$$e^{\boldsymbol{\Omega}(\boldsymbol{\omega})\theta} = \cos\left(\|\boldsymbol{\omega}\|\theta\right)\boldsymbol{I}_4 + \sin\left(\|\boldsymbol{\omega}\|\theta\right)\frac{\boldsymbol{\Omega}(\boldsymbol{\omega})}{\|\boldsymbol{\omega}\|}.$$

Suppose at the initial time, $t_0 = 0$, $\boldsymbol{q}(t_0) = 1\langle 0\rangle$. Integrate equation (1) with the following sequence of inputs:

1. Rotate about the body x-axis for $1/4$ s at π rad s^{-1}.
2. Rotate about the body z-axis for $1/4$ s at π rad s^{-1}.
3. Rotate about the body x-axis for $1/4$ s at $-\pi$ rad s^{-1}.
4. Rotate about the body z-axis for $1/4$ s at $-\pi$ rad s^{-1}.

Report the rotation $\boldsymbol{q}(1)$ at the end of this motion, i.e., after 1s has elapsed. Does it hold that $\boldsymbol{q}(1) = \boldsymbol{q}(0)$? Discuss why this is true or false.

Question 2 .... *35 (main) + 0 (bonus) points*

An input file `input` is provided that contains a list of IMU readings that I collected from my cell phone using HyperIMU. Each line of this file is the IMU data (approximately) $\delta t = 0.01$ second apart, has a comma separated format and contains the following information in exactly the given order:

$$t, m_x, m_y, m_z, \omega_{y,x}, \omega_{y,y}, \omega_{y,z}, a_x, a_y, a_z,$$

where t is the time, $a_\bullet$ and $m_\bullet$ are respectively the components of the linear acceleration and magnetic field measurements $\boldsymbol{a}$ and $\boldsymbol{m}$, expressed in the body-frame. Finally, $\omega_{y,\bullet}$ are the components of $\boldsymbol{\omega}_y$, which is the measured angular velocity through the gyroscope.

We will implement the Mahony filter so that we obtain a good estimate of the orientation of my cell phone for the duration of the time I collected the data in `input`. The Mahony filter is characterized by the following equations

$$\begin{aligned}
\dot{\hat{\boldsymbol{q}}} &= \frac{1}{2}\hat{\boldsymbol{q}} \circ \boldsymbol{p}\left(\boldsymbol{\omega}_y - \hat{\boldsymbol{b}} + k_p\boldsymbol{\omega}_{\text{mes}}\right) =: \frac{1}{2}\hat{\boldsymbol{q}} \circ \boldsymbol{p}(\boldsymbol{u}), \\
\dot{\hat{\boldsymbol{b}}} &= -k_I\boldsymbol{\omega}_{\text{mes}}, \\
\boldsymbol{\omega}_{\text{mes}} &= k_a\boldsymbol{v}_a \times \hat{\boldsymbol{v}}_a + k_m\boldsymbol{v}_m \times \hat{\boldsymbol{v}}_m,
\end{aligned} \tag{2}$$

Any estimated quantity has a *hat* symbol over it. For example, $\hat{\boldsymbol{b}}$ denotes the estimate of the gyroscope bias vector. The normalized acceleration and the normalized magnetic north vector are respectively given by $\boldsymbol{v}_a$ and $\boldsymbol{v}_m$ so that $\|\boldsymbol{v}_a\| = \|\boldsymbol{v}_m\| = 1$. Their estimated counterparts are denoted $\hat{\boldsymbol{v}}_a$ and $\hat{\boldsymbol{v}}_m$. The nonnegative gains k_p, k_I, k_a, and k_m are to be determined. Note that the first equation in equations (2) may be expressed in terms of the rotation matrix $\hat{\boldsymbol{R}}$ associated with the quaternion $\hat{\boldsymbol{q}}$ as follows

$$\dot{\hat{\boldsymbol{R}}} = \hat{\boldsymbol{R}}\left(\boldsymbol{\omega}_y - \hat{\boldsymbol{b}} + k_p\boldsymbol{\omega}_{\text{mes}}\right)_\times =: \hat{\boldsymbol{R}}\boldsymbol{u}_\times. \tag{3}$$

The question is, how does one compute the estimates $\hat{\boldsymbol{v}}_a$ and $\hat{\boldsymbol{v}}_m$? We go back to the definition of $\boldsymbol{v}_a$ in order to determine this. Note that $\boldsymbol{a}$ is the linear acceleration of the rigid-body as expressed in the body frame. If we denote the linear acceleration of the rigid-body, expressed in the inertial frame by $\boldsymbol{a}_0$, then we have the relationship

$$\boldsymbol{v}_a \triangleq \frac{\boldsymbol{a}}{\|\boldsymbol{a}\|} = \boldsymbol{R}^\top \frac{\boldsymbol{a}_0}{\|\boldsymbol{a}_0\|}.$$

The estimated counterpart of this vector is defined analogously, using the estimated orientation, that is,

$$\hat{\boldsymbol{v}}_a = \hat{\boldsymbol{R}}^\top \frac{\boldsymbol{a}_0}{\|\boldsymbol{a}_0\|}.$$

Let's take the time-derivative of this, substitute from equation (3), and live with the consequences:

$$\dot{\hat{\boldsymbol{v}}}_a = \dot{\hat{\boldsymbol{R}}}^\top \frac{\boldsymbol{a}_0}{\|\boldsymbol{a}_0\|} = -\boldsymbol{u}_\times \hat{\boldsymbol{R}}^\top \frac{\boldsymbol{a}_0}{\|\boldsymbol{a}_0\|} = -\boldsymbol{u}_\times \hat{\boldsymbol{v}}_a. \tag{4}$$

This equation must be integrated in order to maintain the estimate $\hat{\boldsymbol{v}}_a$ of the normalized acceleration vector, $\boldsymbol{v}_a$, expressed in body frame. An analogous construction shows that we also must integrate the equation

$$\dot{\hat{\boldsymbol{v}}}_m = -\boldsymbol{u}_\times \hat{\boldsymbol{v}}_m \tag{5}$$

so as to maintain an estimate of the normalized magnetic north vector $\boldsymbol{v}_m$, expressed in the body frame.

Remark 1. *Typically, the magnetometer reading $\boldsymbol{m}$ has a component that points towards world's core, i.e., along the gravitational acceleration vector. Make sure to subtract off this component when constructing the vector $\boldsymbol{v}_m$. Otherwise, your estimation will suffer!*

Now, from problem 1, we know how to integrate the first of equations (2). The second equation is straightforward to integrate

$$\hat{\boldsymbol{b}}_{k+1} = \hat{\boldsymbol{b}}_k - k_I \boldsymbol{\omega}_{\text{mes}} \delta t.$$

Finally, we have to integrate equations (4, 5) so that we can compute the estimator input $\boldsymbol{\omega}_{\text{mes}}$. Again, assuming that $\boldsymbol{u}$ is constant in the selected short integration step, we can integrate these equations in closed-form to obtain

$$\hat{\boldsymbol{v}}_{a,k+1} = e^{-\boldsymbol{u}_\times \delta t} \hat{\boldsymbol{v}}_{a,k}, \qquad \hat{\boldsymbol{v}}_{m,k+1} = e^{-\boldsymbol{u}_\times \delta t} \hat{\boldsymbol{v}}_{m,k}.$$

where the exponential map $\mathfrak{so}(3) \xmapsto{\exp} \boldsymbol{SO}(3)$, is given by the Rodrigues formula:

$$e^{\boldsymbol{u}_\times \theta} = \boldsymbol{I}_3 + \sin\left(\|\boldsymbol{u}\|\theta\right) \frac{\boldsymbol{u}_\times}{\|\boldsymbol{u}\|} + \left(1 - \cos\left(\|\boldsymbol{u}\|\theta\right)\right) \frac{\boldsymbol{u}_\times^2}{\|\boldsymbol{u}\|^2}.$$

Process the input file `input` using your Mahony filter. Compute the amount of rotation $\theta(t)$ as a function of time, taking $\theta(0) = 0$. Plot this with respect to time and then find the total amount my cell phone has rotated during the time I collected this data.

Remark 2. *For some nominal values that you can set for the gains k_p and k_I, please see Section VI of [4]. The remaining two gains k_a and k_m are supposed to be set relative to each other. If there is a lot of magnetic interference you ought to have $k_m \ll k_a$. Conversely, if the motion undergoes a lot of acceleration, you ought to have $k_m \gg k_a$.*

Question 3 .... *20 (main) + 0 (bonus) points*

Set up HyperIMU or SensorLog depending on whether you are using Android or iOS and start collecting your own real-time IMU data from your phone. Set the sampling time to the fastest that your phone can handle. For example, I was able to collect data every 10 ms, however, you may be able to collect faster or slower depending on the hardware of your phone. Collect the data at the fastest rate that you can. Call your Mahony filter online, that is, have it estimate the orientation of your device as you move it around.

(a) [7 points] Find the roll-pitch-yaw angles from your estimated orientation, $\hat{\boldsymbol{q}}$, and plot them with respect to time.

(b) [7 points] Record all the necessary quantities to compute the estimate of the error the Mahony filter is making, given in eqn. (6), and plot it with respect to time.

$$E_{\text{mes}} = 1 - \boldsymbol{v}_a \cdot \hat{\boldsymbol{v}}_a + 1 - \boldsymbol{v}_m \cdot \hat{\boldsymbol{v}}_m. \tag{6}$$

(c) [6 points] Record the gyroscope bias estimate and plot it with respect to time.

Question 4 .... *25 (main) + 5 (bonus) points*

We will implement two more orientation estimators and compare their performances against the Mahony filter that we've implemented in problems 1 through 3.

(a) [5 points] Implement a naïve orientation estimator that integrates raw gyroscope measurements. The naïve estimator takes the gyroscope measurement $\boldsymbol{\omega}_y$ and estimates the orientation $\hat{\boldsymbol{q}}$ by integrating the differential equation in (1), with $\boldsymbol{\omega}$ replaced by $\boldsymbol{\omega}_y$.

(b) [5 points] Implement the triad (Tri-axial Attitude Determination) method, first described in [2] to algebraically estimate an attitude represented as a direction cosine matrix from two orthogonal vector observations.

Given two non-parallel reference *unit vectors* $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ and their corresponding *unit vectors* $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$, it is required to find an orthogonal matrix $\boldsymbol{R}$ satisfying $\boldsymbol{R}\boldsymbol{w}_i = \boldsymbol{v}_i$ for $i = 1, 2$.

Two vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ define an orthogonal coordinate system with the **normalized** basis vectors $\boldsymbol{q}$, $\boldsymbol{r}$, and $\boldsymbol{s}$ as the following triad:

$$\boldsymbol{q}_r = \boldsymbol{v}_1, \qquad \boldsymbol{r}_r = \frac{\boldsymbol{v}_1 \times \boldsymbol{v}_2}{\|\boldsymbol{v}_1 \times \boldsymbol{v}_2\|}, \qquad \boldsymbol{s}_r = \boldsymbol{r}_r \times \boldsymbol{q}_r.$$

These are represented as vectors to build an appropriate *reference frame* $\boldsymbol{M}_r$:

$$\boldsymbol{M}_r = \begin{bmatrix} \boldsymbol{s}_r & \boldsymbol{r}_r & \boldsymbol{q}_r \end{bmatrix}.$$

Similarly, at any given time, two measured vectors in the body frame $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ determine the 3-by-3 body matrix $\boldsymbol{M}_b$:

$$\boldsymbol{M}_b = \begin{bmatrix} \boldsymbol{s}_b & \boldsymbol{r}_b & \boldsymbol{q}_b \end{bmatrix},$$

where, like the first triad, the second triad is built as

$$\boldsymbol{q}_b = \boldsymbol{w}_1, \qquad \boldsymbol{r}_b = \frac{\boldsymbol{w}_1 \times \boldsymbol{w}_2}{\|\boldsymbol{w}_1 \times \boldsymbol{w}_2\|}, \qquad \boldsymbol{s}_b = \boldsymbol{r}_b \times \boldsymbol{q}_b.$$

The attitude (rotation) matrix $\boldsymbol{R} \in \mathbf{SO}(3)$ defines the coordinate transformation: $\boldsymbol{R}\boldsymbol{M}_b = \boldsymbol{M}_r$ and solving for $\boldsymbol{R}$ we obtain

$$\boldsymbol{R} = \boldsymbol{M}_r \boldsymbol{M}_b^\top.$$

For estimations using an IMU, we identify two main reference vectors: gravity $\boldsymbol{g}^\star$ and magnetic field $\boldsymbol{m}^\star = \begin{bmatrix} m_x & m_y & m_z \end{bmatrix}^\top$. A common convention sets the *gravity vector* equal to 0 along the $\boldsymbol{x}-$ and $\boldsymbol{y}-$axes, and equal to ~ 9.81 along the $\boldsymbol{z}-$axis. This assumes the direction of gravity is parallel to the vertical axis. Because triad uses normalized vectors, the $\boldsymbol{z}-$axis turn out to be equal to 1: $\boldsymbol{g}^\star = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^\top$.

The *magnetic field* is defined from the geographical position of the measurement. Using the World Magnetic Model, we can estimate the magnetic field elements of our location on a given date.

Boise State University is located at 43.603600° N and 116.208710° W with an altitude of 835 m. On February 27, 2023 its magnetic elements were obtained as

$$\boldsymbol{m}^\star = \begin{bmatrix} 4525.28449 & 19699.18982 & -47850.850686 \end{bmatrix}^\top \text{nT}.$$

Again, triad works with normalized vectors, so the reference magnetic vector becomes

$$\boldsymbol{m}^\star = \begin{bmatrix} 0.087117 & 0.37923 & -0.92119 \end{bmatrix}^\top.$$

Both normalized vectors $\boldsymbol{v}_1 = \boldsymbol{g}^\star$ and $\boldsymbol{v}_2 = \boldsymbol{m}^\star$ build the *reference triad.*

Then, we have to measure their equivalent vectors, for which we use the accelerometer to obtain $\boldsymbol{a} = \begin{bmatrix} a_x & a_y & a_z \end{bmatrix}^\top$ and the magnetometer for $\boldsymbol{m} = \begin{bmatrix} m_x & m_y & m_z \end{bmatrix}^\top$. Both measurement vectors, $\boldsymbol{w}_1 = \boldsymbol{a}$ and $\boldsymbol{w}_2 = \boldsymbol{m}$ are also normalized, meaning $\|\boldsymbol{w}_1\| = \|\boldsymbol{w}_2\| = 1$, so they can build the *body's measurement triad* $\boldsymbol{M}_b$.

(c) [10 points] Process the input file `input` from problem 2 again with the naïve estimator and the triad method and plot the amount of rotation, $\theta(t)$, with respect to time. Also compute the total amount my cell phone has rotated during the time I collected this data using the naïve estimator and the triad method.

(d) [5 points] Repeat problem 3 using the naïve estimator and the triad method. Since none of these methods keeps an estimate of the gyroscope bias, you can ignore part (c) of problem 3 for this question.

(e) [5 points (bonus)] Provide a discussion about which of the estimation methods works better for the `input` file that I have provided. Can you imagine a scenario in which the performances of the various estimators would be very different from what you observed for this dataset? What would the characteristics of such a dataset be? Can you create a such a motion with your cell phone and show such behavior?

References

[1] Mahony Orientation Filter. URL https://ahrs.readthedocs.io/en/latest/filters/mahony.html.

[2] H. D. Black. A passive system for determining the attitude of a satellite. *AIAA journal*, 2(7):1350–1351, 1964.

[3] S. Campisi. HyperIMU. URL https://github.com/JohannesFriedrich/HypeRIMU.

[4] R. Mahony, T. Hamel, and J.-M. Pflimlin. Nonlinear complementary filters on the special orthogonal group. *IEEE Transactions on automatic control*, 53(5):1203–1218, 2008.

[5] B. Thomas. SensorLog. URL http://sensorlog.berndthomas.net/.

[6] J. R. Wertz. *Spacecraft attitude determination and control*, volume 73. Springer Science & Business Media, 2012.