

## Overview

High-level autonomy requires confidence in low-level control and estimation. Autopilot firmware can use an inertial measurement unit (IMU) with three orthogonal rate gyroscopes and three orthogonal accelerometers to perform estimation. If sensors were perfect, the estimator would simply integrate the gyro measurements to obtain an estimate of the attitude. However, gyros tend to drift over time due to bias. The accelerometer can be used to correct the drift in the roll and pitch axes, however yaw bias is unobservable with an IMU alone. This document explores various schemes and implementations of attitude estimation using an IMU, with occasional external attitude updates from vision, motion capture, or some other source. Another good reference is [1].

## IMU Model

With the advent of micro electro-mechanical systems (MEMS), the scale of IMUs has decreased significantly over the years. In particular, the spread of small MEMS IMUs is in part due to the proliferation of smart phones and wearable electronics.

To understand the IMU model, consider a rigid body  $\{B\}$  with a position  $\mathbf{r}$  expressed in an inertial frame  $\{A\}$ . An IMU is ‘strapped down’ to the origin of the rigid body and is axis aligned with it. In other words, the IMU sensor frame  $\{S\}$  is identified with the body frame  $\{B\}$ . Unless otherwise stated, we will make use of the East-North-Up (ENU) inertial frame with its corresponding Front-Left-Up (FLU) body frame. The rotation matrix  $R_B^A$  is the local-to-global rotation of  $\{B\}$  w.r.t  $\{A\}$  and it takes data from  $\{B\}$  into  $\{A\}$ . The bases (axes) of  $\{A\}$  are written  $\{\mathbf{e}_i\}$ .

With this problem geometry, we write the measurements received by the IMU below.

### Accelerometer

Accelerometers do not measure gravity. Instead, an accelerometer measures the *specific force* (i.e.,  $\mathbf{F}/m$ ) that prevents free fall. Thus, an accelerometer free falling in a vacuum would measure 0, while an accelerometer sitting on a table would measure the normal force that is preventing free fall. The measurement equation is

$$\mathbf{a} = (R_B^A)^\top (\ddot{\mathbf{r}} + g\mathbf{e}_3) + \boldsymbol{\nu} + \mathbf{b}_a, \quad (1)$$

where  $g \approx 9.81 \text{ m/s}^2$ ,  $\boldsymbol{\nu}$  is zero-mean Gaussian noise with variance  $\Sigma_a^2$  and  $\mathbf{b}_a$  is a bias term. Note that it may seem contrary to have *plus*  $g\mathbf{e}_3$ , but this is correct since we are working in an ENU/FLU frame, where gravity is actually  $-g\mathbf{e}_3$ ; therefore, we are actually *subtracting* the gravity acceleration vector.

MEMS accelerometers often have a sample rate of 100 Hz to 1 kHz.

### Rate Gyro

The measured angular rate  $\boldsymbol{\omega} = [p \quad q \quad r]^\top$  is the body rate of the vehicle with respect to the inertial frame, expressed in the body frame and is written as

$$\boldsymbol{\omega} = \boldsymbol{\omega}_{\text{true}} + \boldsymbol{\eta} + \mathbf{b}, \quad (2)$$

where  $\boldsymbol{\eta}$  is zero-mean Gaussian noise with variance  $\Sigma_\omega^2$  and  $\mathbf{b}$  is constant or slowly time-varying bias (i.e.,  $\dot{\mathbf{b}} \approx 0$ ).

Gyros typically have a sample rate of 500 Hz, 1 kHz, or 8 kHz. They are great at capturing high-frequency dynamics and are incredibly useful, especially over short time windows. However, due to the low-frequency bias  $\mathbf{b}$ , simply integrating gyros over long timescales is unwise (e.g., more than tens of seconds).

### Allan Variance

### AR(n) Model

## Complementary Filter

One of the most basic estimators is the *complementary filter*, also known as the *balance filter* (Shane Colton, 2007). The motivating principle is that gyros are good at capturing high-frequency dynamics, but have a low-frequency drift. On the otherhand, accelerometers tend to have high-frequency noise (e.g., from non-smooth movement). Further, when the system that the IMU is attached to has actuators (like a multirotor),

## Theory

As indicated above, complementary filters are useful for fusing independent noisy measurements of the same signal with complementary spectral characteristics. Here, we give an overview of the theory of complementary filters and derive a linear estimator for a kinematic dynamic process with constant bias. This theoretical overview follows the discussion in Appendix A of Mahony et al. [2].

### Preliminary Discussion: Linear system with no dynamics

Consider a time-varying signal  $x(t)$  we wish to observe. We have two sensors  $y_1$  and  $y_2$  that measure the signal with additive noise

$$\begin{aligned} y_1(t) &= x(t) + \mu_1(t) & \mu_1 &\rightarrow \text{high-frequency noise/disturbance} \\ y_2(t) &= x(t) + \mu_2(t) & \mu_2 &\rightarrow \text{low-frequency noise/disturbance} \end{aligned} \quad (3)$$

Given that we know the noises are complementary in a spectral sense, we can design a pair of complementary transfer functions,  $F_1(s) + F_2(s) = 1$ , to filter out the noise while allowing the signal to pass. Filtering each measurement through its corresponding transfer function results in the estimate

$$\begin{aligned} \hat{x}(s) &= F_1(s)y_1(s) + F_2(s)y_2(s) \\ &= F_1(s)[x(s) + \mu_1(s)] + F_2(s)[x(s) + \mu_2(s)] \end{aligned} \quad (4)$$

$$\begin{aligned} &= (F_1(s) + F_2(s))x(s) + F_1(s)\mu_1(s) + F_2(s)\mu_2(s) \\ &= x(s) + F_1(s)\mu_1(s) + F_2(s)\mu_2(s), \end{aligned} \quad (5)$$

where we construct  $F_1(s)$  to be low-pass and  $F_2(s)$  to be high-pass to appropriately filter out the known spectral content of the disturbances. Notice that  $x(s)$  is all-pass as desired. Recall that a first-order low-pass filter has the frequency response

$$H_{\text{LPF}}(s) = \frac{\omega_c}{s + \omega_c} = \frac{1}{\tau s + 1}; \quad \tau = \frac{1}{\omega_c}. \quad (6)$$

The complement of a LPF is a high-pass filter

$$H_{\text{HPF}}(s) = 1 - H_{\text{LPF}}(s) = \frac{s}{s + \omega_c} = \frac{\tau s}{\tau s + 1}. \quad (7)$$

Thus, by letting  $F_1(s) = H_{\text{LPF}}(s)$  and  $F_2(s) = H_{\text{HPF}}(s)$ , we obtain the desired frequency characteristics.

### Sampled-Data Implementation

While equation (5) is useful for analysis, we would like to be able to implement this filter as an algorithm in a digital computer. This requires implementing equation (4), which uses the measurements  $y_1$  and  $y_2$ . Here, we will discuss two<sup>1</sup> sampled-data implementations of the vanilla linear complementary filter. Both versions start with using the inverse Laplace transform to rewrite equation (4) as differential equations. Once in the time domain, we can perform a zero-order hold on the solution to the differential equation. Alternatively, an approximation method can be used such as Euler's method, avoiding the necessity of solving the differential equation.

First, we start with the inverse Laplace transform, starting from equation (4)

$$\begin{aligned} \hat{x}(s) &= \frac{\omega_c}{s + \omega_c} y_1(s) + \frac{s}{s + \omega_c} y_2(s) \\ (s + \omega_c)\hat{x}(s) &= \omega_c y_1(s) + s y_2(s) \\ s\hat{x}(s) &= s y_2(s) + \omega_c (y_1(s) - \hat{x}(s)), \end{aligned}$$

which becomes

$$\dot{\hat{x}}(t) = \dot{y}_2(t) + \omega_c (y_1(t) - \hat{x}(t)).$$

Note that how this is written, it appears to be a pair of coupled first-order differential equations. However, we do not actually know how  $y_2(t)$  evolves, and we are interested in the value of the derivative at time  $t$ , which we can obtain via numerical differentiation. For clarity, we will define  $y_2^\Delta(t) := \dot{y}_2(t)$ , leaving us with

$$\dot{\hat{x}}(t) = -\omega_c \hat{x}(t) + \omega_c y_1(t) + y_2^\Delta(t). \quad (8)$$

**1) ZOH.** The zero-order hold method involves solving the differential equation. This is not difficult and is left as an exercise to the reader (see Chapter 4.1 of Beard [3]).

<sup>1</sup>A third way uses the bilinear transform to move from the Laplace domain to the Z-domain, followed by the inverse Z transform.

**2) Approximation via Euler's Method.** In this case, we approximate the time derivative using the limit definition, giving us

$$\begin{aligned}\frac{\hat{x}_i - \hat{x}_{i-1}}{\Delta t} &= -\omega_c \hat{x}_i + \omega_c y_{1,i} + \frac{y_{2,i} - y_{2,i-1}}{\Delta t} \\ \hat{x}_i &= -\Delta t \omega_c \hat{x}_i + \Delta t \omega_c y_{1,i} + y_{2,i} - y_{2,i-1} + \hat{x}_{i-1} \\ (1 + \Delta t \omega_c) \hat{x}_i &= \Delta t \omega_c y_{1,i} + y_{2,i} - y_{2,i-1} + \hat{x}_{i-1} \\ \hat{x}_i &= \frac{1}{1 + \Delta t \omega_c} (\Delta t \omega_c y_{1,i} + y_{2,i} - y_{2,i-1} + \hat{x}_{i-1}) \\ &= \frac{\Delta t \omega_c}{1 + \Delta t \omega_c} y_{1,i} + \frac{1}{1 + \Delta t \omega_c} (y_{2,i} - y_{2,i-1}) + \frac{1}{1 + \Delta t \omega_c} \hat{x}_{i-1}.\end{aligned}$$

We can write this compactly as

$$\hat{x}_i = \alpha \hat{x}_{i-1} + \alpha (y_{2,i} - y_{2,i-1}) + (1 - \alpha) y_{1,i}, \quad (9)$$

where

$$\alpha := \frac{1}{1 + \Delta t \omega_c} = \frac{\tau}{\tau + \Delta t}; \quad 1 - \alpha := \frac{\Delta t \omega_c}{1 + \Delta t \omega_c} = \frac{\Delta t}{\tau + \Delta t}; \quad 0 \leq \alpha \leq 1.$$

Note that we could have gotten to the same equation by deriving the  $\alpha$ -LPF and  $\alpha$ -HPF variants and summing them.

### Example

In this example, we use a sample period of  $T_s = 0.001$  s and take  $x(t) = \cos(2\pi f_x t)$ , where  $f_x = 5$  Hz. The disturbances  $\mu_1$  and  $\mu_2$  are specified as sinusoids with frequencies 20 times higher and lower, respectively. The results are shown in Fig. 1b for filters designed with a cut-off frequency of  $f_c = 1$  Hz (top) and  $f_c = 10$  Hz (bottom). The frequency responses of the two filter designs are shown in Fig. 1c, in terms of the complementary filters  $F_1(s)$  and  $F_2(s)$ .

Note how varying the cut-off frequency  $f_c$  changes the response of the filter, choosing between rejecting more low-frequency or high-frequency noise. Because of the characteristics of the first-order filters used, the transition band is quite wide. This results in noise leakage into the estimate when the frequency of the true signal is close to the frequencies of the noise. By increasing the ratio of the frequencies, we are able to reconstruct  $x(t)$  perfectly.

Specifically, when the cut-off frequency is close to the low-frequency disturbance ( $\mu_2$ ), high-frequency noise is heavily attenuated, but low-frequency noise is not as much. The opposite holds true when the cut-off frequency is close to the high-frequency disturbance ( $\mu_1$ ). This translates to a choice of  $\alpha$  small ( $f_c$  large) allowing more high-frequency noise, while  $\alpha$  large ( $f_c$  small) allows more low-frequency noise.

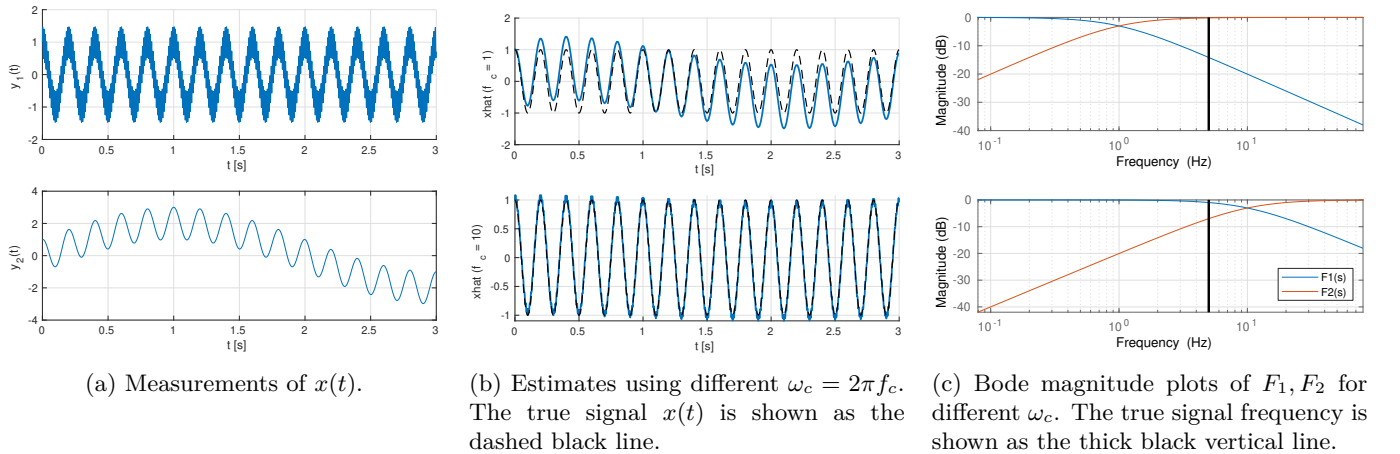


Figure 1: Results from an example complementary filter applied to simulated data.

## Mahony's Nonlinear Complementary Filter

One of the more popular estimators is Mahony's nonlinear complementary filter on  $SO(3)$  [2].

$$\dot{\hat{\mathbf{q}}} = \frac{1}{2} \hat{\mathbf{q}} \otimes \bar{\boldsymbol{\omega}} \quad (10)$$

$$\dot{\bar{\boldsymbol{\omega}}} = (\boldsymbol{\omega} - \mathbf{b}) + k_P \boldsymbol{\omega}_{\text{err}} \quad (11)$$

$$\dot{\mathbf{b}} = -k_I \boldsymbol{\omega}_{\text{err}} \quad (12)$$

$$\boldsymbol{\omega}_{\text{err}} = \sum v_i \times \hat{\mathbf{v}}_i \quad (13)$$

$$\tilde{\mathbf{q}} = \hat{\mathbf{q}}^{-1} \otimes \mathbf{q}(\mathbf{a}) \quad (14)$$

$$\boldsymbol{\omega}_{\text{err}} = 2q_w \mathbf{q}_{\text{vec}} \quad (15)$$

## References

- [1] M. Kok, J. D. Hol, and T. B. Schön, “Using Inertial Sensors for Position and Orientation Estimation,” *Found. Trends Signal Process.*, vol. 11, no. 1-2, pp. 1–153, Nov. 2017. [Online]. Available: <https://arxiv.org/pdf/1704.06053.pdf>
- [2] R. Mahony, T. Hamel, and J. Pflimlin, “Nonlinear Complementary Filters on the Special Orthogonal Group,” *IEEE Transactions on Automatic Control*, vol. 53, no. 5, pp. 1203–1218, June 2008.
- [3] R. W. Beard, “Robot soccer notes,” Brigham Young University, Tech. Rep., March 2017. [Online]. Available: [http://rwbclasses.groups.et.byu.net/lib/exe/fetch.php?media=robot\\_soccer:material:robot\\_soccer\\_notes.pdf](http://rwbclasses.groups.et.byu.net/lib/exe/fetch.php?media=robot_soccer:material:robot_soccer_notes.pdf)