ASD to State-Space Example: Sample Notebook Result

Jay A. Farrell

August 12, 2025

Abstract

This article presents and discusses example results of the Python Jupyter notebook:

```
Demo_FirstOrderGM_AccelMarbleSlab.ipynb
```

It should exist in the same directory as this pdf file.

1 Problem Setup

The blue dots in Fig. 1 show samples of the Allan Standard Deviation (ASD) curve for an accelerometer for delays $\tau \in [0.01, 500]$ seconds. The sample frequency is $F_s = 100$ Hz.

The optimal estimator that we are building requires a finite-dimensional, discrete-time state-space model for the stochastic error in the IMU output that has the form

$$\mathbf{x}(k+1) = \mathbf{\Phi}\,\mathbf{x}(k) + \mathbf{\Gamma}\,\boldsymbol{\omega}(k) \quad (1)$$
$$z(k) = \mathbf{C}\,\mathbf{x}(k) + \nu(k), \quad (2)$$

where $\mathbf{x} \in \Re^n$, $z(k)$ is scalar, and the matrices have appropriate dimensions. Both $\boldsymbol{\omega} \sim N(\mathbf{0}, \mathbf{Q}_\omega)$ and $\nu \sim N(0, Q_\nu)$ are white and Gaussian.

There is no finite-dimensional model that can match the ASD curve for all delays τ. The approximate fit to the curve can be improved by increasing n; however [1]:

- The computational load increase with the cube of the number of states in the estimator. Because the IMU has six out puts, six error models are required, so the selection of n has computation impact proportional to $(6n)^3$.
- Adding states can have a negative impact on observability that can negatively impact performance.

The purpose of the state-space model is to communicate to the estimator the magnitude and correlation properties of the IMU output stochastic error sequence of the time intervals relevant to the estimator. The lowest order state-space model that can do the job should be selected.

2 Specifications

For this example the goal is to match (or be slightly above) the ASD graph for delays $\tau \in [T_s, 100]$ seconds, where $T_s = \frac{1}{F_s}$.

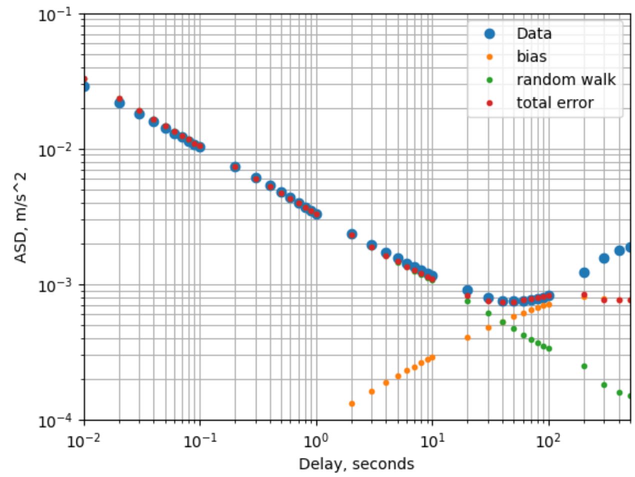


Figure 1: Allan Standard Deviation Plot.

3 Selection of AV Parameters

The value of N is read from the ASD graph as the value at $\tau = 1$ second: $N = 3.3e-3$ m/s/s/rtHz = m/s/rtsec. The method to compute Q_ν from N are documented in the python code.

This example will use a state-space model order $n = 1$ where the single state element will be denoted by b and referred to as a bias. The model has simplifies to

$$b(k+1) = \mu\,b(k) + \omega(k) \quad (3)$$
$$z(k) = b(k) + \nu(k). \quad (4)$$

The method to compute the model parameters μ and Q_ω based on the choice of the bias instability parameter B and the parameter T_p are documented in the python code. The values of B and the parameter T_p are adjusted manually to achieve the desired ASD shape in the region where the curve flattens. For the present example, the bias instability parameter is selected as $B = 9.0e-4$ m/s/s and the desired delay at which the ASD plot of the Gauss-Markov model should have its peak is $Tp = 300$ seconds.

4 Results

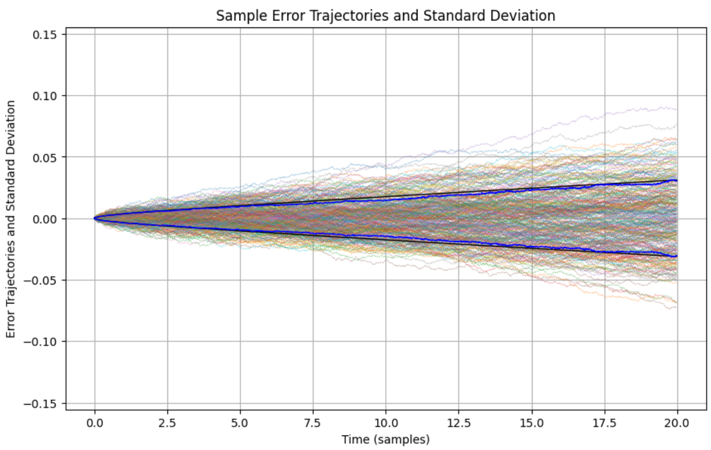
The Jupyter notebook outputs the following:

- Continuous-time model is (units depend on context):
  z(t) = b(t) + n(t),
  db(t)/dt = -6.3000e-03 * b(t) + w(t),
  where n(t) is white noise with PSD $Sn = 1.0890e-05$ and w(t) is the bias process noise with PSD $Sw = 2.6783e-08$ The steady-state covariance of the bias process is Pb_ss_c = 2.1256e-06.

- Discrete-time model is (units depend on context):
  z[k] = b[k] + n[k],
  b[k+1] = 0.9999 * b[k] + w[k],
  where n[k] is white noise with covariance Qn = 1.0890e-03 (std 3.3000e-02) and w[k] is the bias process with covariance Qw = 2.6781e-10 (std 1.6365e-05). The steady-state covariance of the bias process is Pb_ss_d = 2.1256e-06.

Note that both models have the same steady-state covariance for the bias.

The discrete-time model is easily simulated to yield data sequences that can be used to check the validity of the error model.



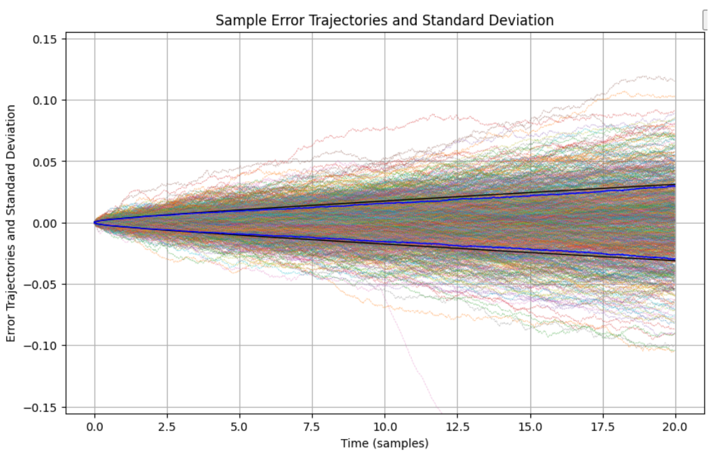


Figure 2: Covariance Check Graphs. Top: One thousand sample error trajectories computed by integrating the output of the state-space error model. Bottom: One thousand sample error trajectories computed by integrating the IMU data after subtracting a prior estimate of the bias. Black solid curve is the theoretical standard deviation from the state-space model. Blue solid curve is the sample standard deviation at each time instant over the sample trajectories.

4.1 Check: ASD Plot

Fig. 1 shows the ASD graph for the simulate output of the state-space error model as red dots. These closely match the ASD plot for the instrument data for $tau \in [T_s, 100]$ seconds.

The figure also shows the ASD plot for both components b (orange)and ν (green) that are summed to create the overall output z. The value of B was selected to get the appropriate height of the the ASD for b and the value of T_p was adjusted so that the overall plot had the desired shape for $\tau \in [10, 100]$ seconds.

4.2 Check: Covariance

The main purpose of the model is to communicate to the optimal estimation routine the magnitude and correlation structure of the IMU errors. The IMU outputs are integrated to produce angles or velocities.

Therefore, this section compares the growth of the growth of the integrated error in a few ways:

- The error model produces 1000 simulated integrated error trajectories. Each is plotted as a narrow opaque curve in the top portion of Fig. 2. At each time instant the robust standard deviation is computed and plotted as a blue line. The theoretical error covariance is computed using the state-space model and eqn. (4.125) in [2]. This is plotted as the black line. Note that the blue and black lines match closely at all times.

- After removing a constant prior estimate of the bias, the instrument data produces 1000 simulated integrated error trajectories. Each is plotted as a narrow opaque curve in the bottom portion of Fig. 2. At each time instant the robust standard deviation is computed and plotted as a blue line. The theoretical error covariance is computed using the state-space model and eqn. (4.125) in [2]. This is plotted as the black line. Note that the blue and black lines match closely at all times.

References

[1] J. A. Farrell, F. O. Silva, F. Rahman, and J. Wendel, "Inertial measurement unit error modeling tutorial: Inertial navigation system state estimation with real-time sensor calibration," *IEEE Control Systems Magazine*, vol. 42, no. 6, pp. 40–66, 2022.

[2] J. A. Farrell, *Aided Navigation: GPS with High Rate Sensors*. McGraw-Hill, Inc., 2008.