Project Report

Alex Xu

I. Abstract

This report presents a kalman filter used to calibrate the accelerometer of a vehicle with GPS measurements. The model of the accelerometer is presented and the Kalman FIlter is tested in a Monte-Carlo simulation over numerous cycles. Results of the error, orthogonality, and residuals show that the Kalman Filter is working as expected.

II. Introduction

The goal of this project is to calibrate an accelerometer with GPS measurements for a vehicle traveling in one dimension from time $t = [0, 30s]$. The acceleration of the vehicle is:

$$a(t) = sin(wt) \tag{1}$$

- $w = 0.1\frac{rad}{s}$

The accelerometer samples data at a rate of 200 Hz and also experiences additive, white, Gaussian noise $\omega \sim N(0, 0.0004(m/s^2)^2)$ and bias $b_a \sim N(0, 0.01(m/s^2)^2)$. The accelerometer at time t_j can be modeled as:

$$a_c(t_j) = a(t_j) + w(t_j) + b_a \tag{2}$$

The GPS samples position and velocity at a rate of 5 Hz, with measurements synchronized to the accelerometer. The GPS can be modeled as:

$$z_i = \begin{cases} x_i + \eta_1 \\ v_i + \eta_2 \end{cases} \tag{3}$$

- $x_0 \sim N(0, 10(m)^2)$
- $v_0 \sim N(100, 1(m/s)^2)$
- $\begin{bmatrix}\eta_1 \\ \eta_2\end{bmatrix} \sim N(\begin{bmatrix}0 \\ 0\end{bmatrix}, \begin{bmatrix}1(m)^2 & 0 \\ 0 & 4(cm/s)^2\end{bmatrix})$

III. Theory

A. Truth Model

From Eq. (1), the actual position and velocity is:

$$\begin{aligned} v(t) &= v(0) + \frac{a}{w} - \frac{a}{w}cos(wt) \\ p(t) &= p(0) + (v(0) + \frac{a}{w})t - \frac{a}{w^2}sin(wt) \end{aligned} \tag{4}$$

B. Accelerometer Model

From Eq. (2), the position and velocity are found through integrating by an Euler formula where $\Delta t = t_{j+1} - t_j = 0.02s$:

$$\begin{aligned} v_c(t_{j+1}) &= v_c(t_j) + a_c(t_j)\Delta t \\ p_c(t_{j+1}) &= p_c(t_j) + v_c(t_j)\Delta t + a_c(t_j)\frac{\Delta t^2}{2} \end{aligned} \tag{5}$$

The values and statistics of v_0 and p_0 are given in Eq. (3). Plots showing the values of position, velocity, and acceleration are shown in Fig. 1, Fig. 2, and Fig. 3. The plots show the true, model, and GPS values.

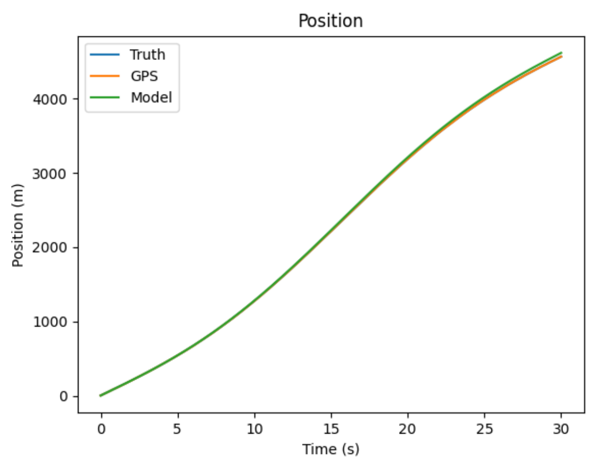

Fig. 1: Plot of Position Showing Truth, Model, and GPS.

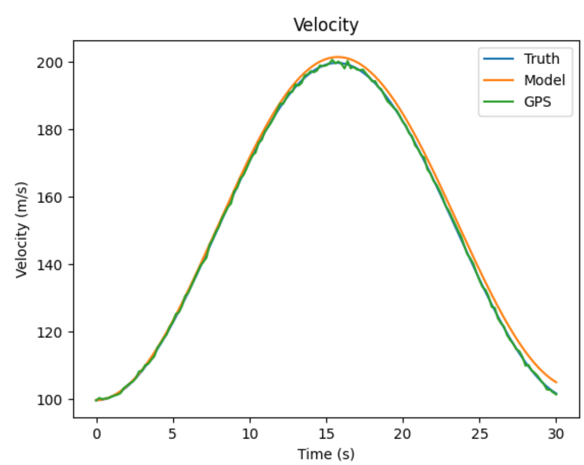

Fig. 2: Plot of Velocity Showing Truth, Model, and GPS.

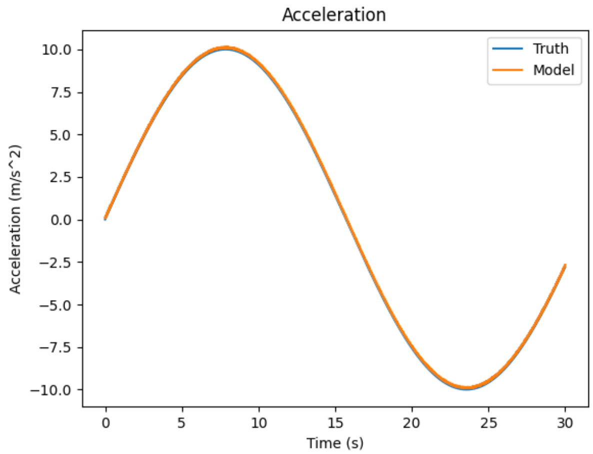

Fig. 3: Plot of Acceleration Showing Truth, Model.

C. Dynamic Model

Assuming that the true acceleration is integrated by the same Euler formula in Eq. (5):

$$\begin{aligned} v_E(t_{j+1}) &= v_E(t_j) + a(t_j)\Delta t \\ p_E(t_{j+1}) &= p_E(t_j) + v_E(t_j)\Delta t + a(t_j)\frac{\Delta t^2}{2} \end{aligned} \tag{6}$$

The values and statistics of v_0 and p_0 are given in Eq. (3). Subtracting Eq. (1), and Eq. (5) from Eq. (2) and Eq. (6) gives the following:

$$\begin{bmatrix} \delta p_E(t_{j+1} \\ \delta v_E(t_{j+1}) \\ b(t_{j+1}) \end{bmatrix} = \begin{bmatrix} 1 & \Delta t & -\frac{\Delta t^2}{2} \\ 0 & 1 & -\Delta t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta p_E(t_j \\ \delta v_E(t_j) \\ b(t_j) \end{bmatrix} - \begin{bmatrix} \frac{\Delta t^2}{2} \\ \Delta t \\ 0 \end{bmatrix} w(t_j) \tag{7}$$

- $\delta p_E(t_0) = p_E(t_0) - p_c(t_0) \sim N(0, M_0^p)$
- $\delta v_E(t_0) = v_E(t_0) - v_c(t_0) \sim N(0, M_0^v)$
- $E[w(t_j)] = 0, E[w(t_j)e(t_l)^T] = W\delta_{j,l}$
- $b(0) \sim N(0, M_0^b)$

Eq. (7) makes the coefficients independent of the acceleration profile.

D. Measurement Equations

The GPS measurements should be slightly modified to work with the Kalman filter. The measurement equations are:

$$\begin{aligned} \delta z^p(t_i) &= \delta p(t_i) + \eta_1(t_i) \\ \delta z^v(t_i) &= \delta v(t_i) + \eta_2(t_i) \end{aligned}$$

- $\delta p(t_i) = p(t_i) - p_c(t_i)$
- $\delta v(t_i) = v(t_i) - v_c(t_i)$

E. Kalman Filter

For a linear system described by the following equations:

$$\begin{aligned} x_{k+1} &= \Phi_k x_k + \Gamma_k w_k,\ x_0 \sim N(\bar{x_0}, M_0) \\ z_k &= H_k x_k + v_k \end{aligned} \tag{8}$$

The Kalman filter equations are defined as[1]:

$$\bar{x}_{k+1} = \Phi_k \hat{x}_k \tag{9}$$

$$M_{k+1} = \Phi_k P_k \Phi_k^T + \Gamma_k W_k \Gamma_k^T \tag{10}$$

$$\hat{x_k} = \bar{x}_k + P_k H_k^T V_k^{-1}(z_k - H_k \bar{x}_k) \tag{11}$$

$$P_k = (M_k^{-1} + H_k^T V_k^{-1} H_k)^{-1} \tag{12}$$

where

- $\bar{x}_{k+1}$ is the propagated estimate at the next stage
- M_{k+1} is the propagated covariance at the next stage
- $\hat{x_k}$ is the updated conditional mean
- P_k is the updated covariance

Based on the dynamics equations in Section III-C, the approximate posteriori conditional mean is defined as:

$$\hat{\delta x}(t_i) = \begin{bmatrix} \hat{\delta p_c}(t_j - \delta p_c(t_j \\ \hat{\delta v_c}(t_j) - \delta v_c(t_j) \\ \hat{b}(t_j) \end{bmatrix} \tag{13}$$

Assuming that $\delta p(t_i) \approx \delta p_E(t_i)$ and $\delta v(t_i) \approx \delta v_E(t_i)$

F. Error Checking

The *a priori* estimation error is defined as:

$$\bar{e}(t_j) = \begin{bmatrix} p(t_j - \bar{p}(t_j) \\ v(t_j) - \bar{v}(t_j) \\ b(t_j - \bar{b}(t_j)) \end{bmatrix} \tag{14}$$

Using a Monte-Carlo simulation, the ensemble average can be found over a large number of realizations N. The ensemble average is defined as:

$$e^{ave}(t_j) = \frac{1}{N} \sum e^l(t_i) \tag{15}$$

where $e^l(t_i)$ is the error from a realization $l \exists [0, N]$. The expected value of $e^{ave}(t_i) \approx 0$ for all values of $t \exists [0, 30]$.

The ensemble average can also be used to calculate the actual error variance P^{ave}:

$$P^{ave} = \frac{1}{N-1} \sum [e^l(t_i) - e^{ave}(t_i)][e^l(t_i) - e^{ave}(t_i)]^T \tag{16}$$

The actual error variance should be close to the covariance matrix P_k computed in the Kalman Filter. Another check is the orthogonality of the error in estimates with the estimate should be close to 0.

$$\frac{1}{N} \sum [e^l(t_i) - e^{ave}(t_i)]\hat{x}(t_i)^T \approx 0 \forall t_i \tag{17}$$

Lastly, the independence of the residuals should also be close to 0. The residual for a realization l is defined as:

$$r^l(t_i) = \begin{bmatrix} \delta z^{p,l}(t_i) - \bar{\delta z}^{p,l}(t_i) \\ \delta z^{v,l}(t_i) - \bar{\delta z}^{v,l}(t_i) \end{bmatrix} \tag{18}$$

The ensemble average for the correlation of two residuals at time t_i and t_m is defined as:

$$\frac{1}{N} \sum r^l(t_i) r^l(t_m) \approx 0 \forall t_m < t_i \tag{19}$$

IV. Algorithm

Since the GPS has a different update rate than the accelerometer, the mean and covariance update functions are only performed when there is a GPS measurement. The Kalman filter procedure is outlined in Algorithm 1

A. Pseudocode

Algorithm 1 Kalman Filter

Input: z_i
Output: $\bar{x}_k$
Initialize: $b_a \sim N(0, M_b^0), p_0 \sim N(\bar{p}_0, M_p^0), v_0 \sim N(\bar{v}_0, M_v^0)$
for t_j ϵ [0, 30] **do**
  if $\exists z_i$ **then**
    Calculate conditional mean $\hat{x}_k$ Eq. (11)
    Calculate covariance P_kEq. (12)
    Propagate mean $\bar{x}_{k+1}$Eq. (9)
    Propagate covariance M_{k+1}Eq. (10)
  else
    Propagate mean $\bar{x}_{k+1}$Eq. (9)
    Propagate covariace M_{k+1}Eq. (10)
  end if
end for

V. Results

The error results of one realization are shown in Fig. 4, Fig. 5, and Fig. 6. The lines of the σ bounds are also shown. The σ bound is calculated using the $\sqrt{P_{ii}}$ of the covariance matrix with ii corresponding to the diagonal index. Fig. 4, Fig. 5, and Fig. 6 show that the error generally stays within the σ bounds with some slight deviations. All the graphs have the error converge to 0.

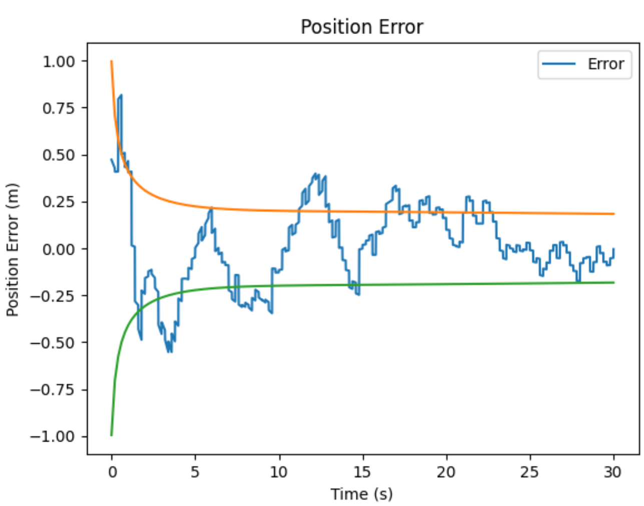

Fig. 4: Position Error of the Kalman Filter. The σ-bounds are plotted in green and orange.

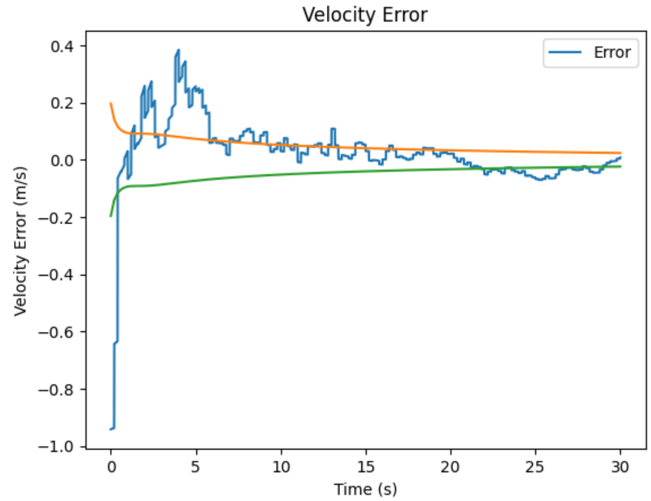

Fig. 5: Velocity Error of the Kalman Filter. The σ-bounds are plotted in green and orange.

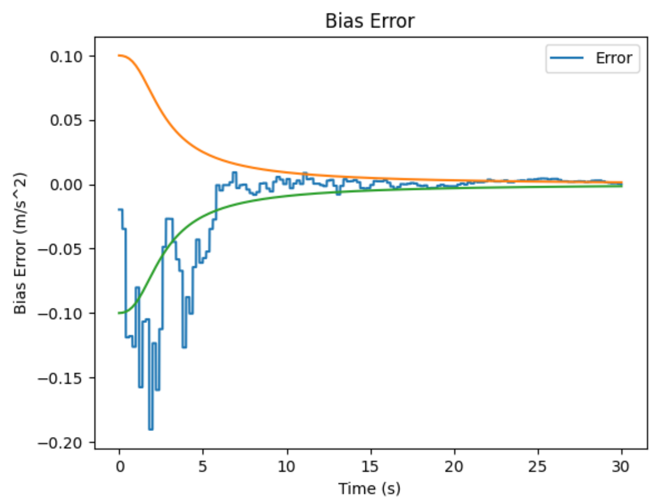

Fig. 6: Bias Error of the Kalman Filter. The σ-bounds are plotted in green and orange.

VI. Performance

A total of 100 realizations were done to test the Kalman filter. Fig. 7, Fig. 8, and Fig. 9 show the average error over the 100 realizations. The graphs all show very small error values over the entire 30-second interval.

Fig. 10 shows the actual error variance. Fig. 11 shows the result of the actual error variance P^{ave} subtracted by the covariance matrix P_i. As expected, the values are very close to 0 for every value in the matrix over the entire interval.

Fig. 12 shows the orthogonality matrix. The results for the orthogonality matrix are a bit noiser, but most variations are small, and every value is close to 0.

Finally, the residual for two time instances t_i and t_m is calculated to be:

$$\frac{1}{N}\sum r^l(t_i)r^l(t_m) = \begin{bmatrix} 0.00481986 & -0.00267533 \\ 0.00481986 & -0.00267533 \end{bmatrix} \approx 0 \tag{20}$$

The norm of this matrix is 0.007796.

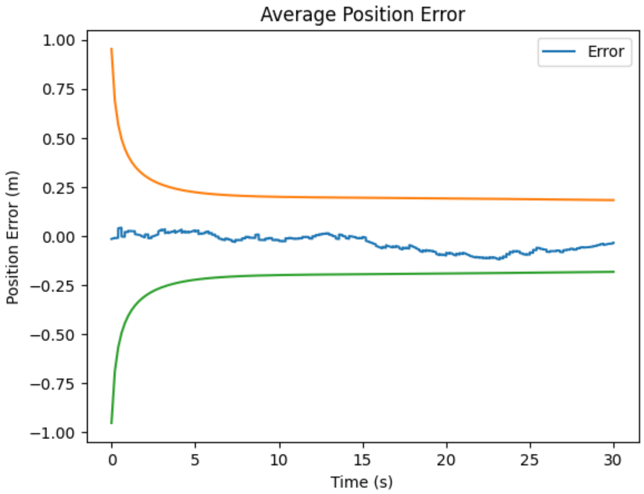

Fig. 7: Average Position Error of the Kalman Filter. The σ-bounds are plotted in green and orange.

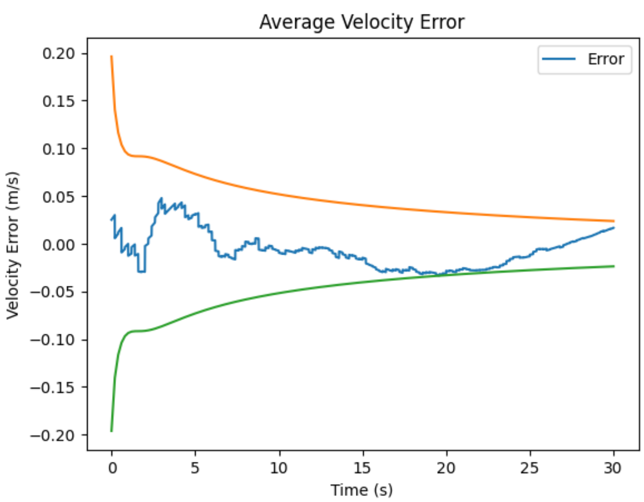

Fig. 8: Average Velocity Error of the Kalman Filter. The σ-bounds are plotted in green and orange.

VII. Conclusion

A Kalman Filter was implemented to calibrate an accelerometer with GPS measurements. Numerous tests were run to show that the Kalman Filter worked as expected. The results show that a Kalman filter can reduce the error in the prediction of the current state.

ACKNOWLEDGMENT

Thank you to Professor Speyer for providing guidance, and instruction for this assignment.

References

[1] Speyer, J.L., Chung, W.H.: Stochastic Processes, Estimation, and Control. Society for Industrial and Applied Mathematics (2008). https://doi.org/10.1137/1.9780898718591, `https://epubs.siam.org/doi/abs/10.1137/1.9780898718591`

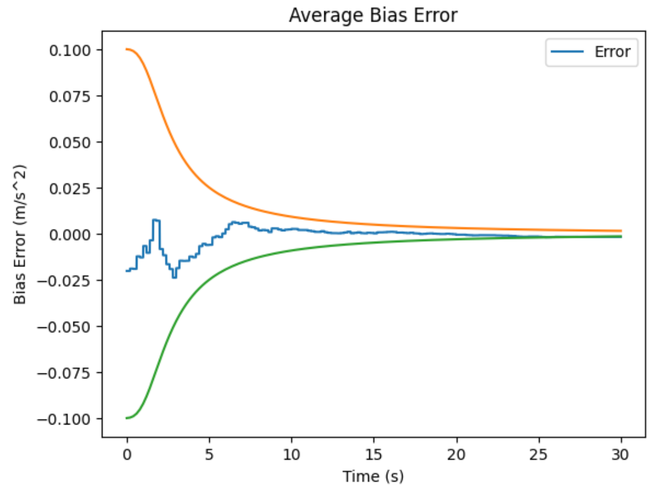

Fig. 9: Average Bias Error of the Kalman Filter. The σ-bounds are plotted in green and orange.

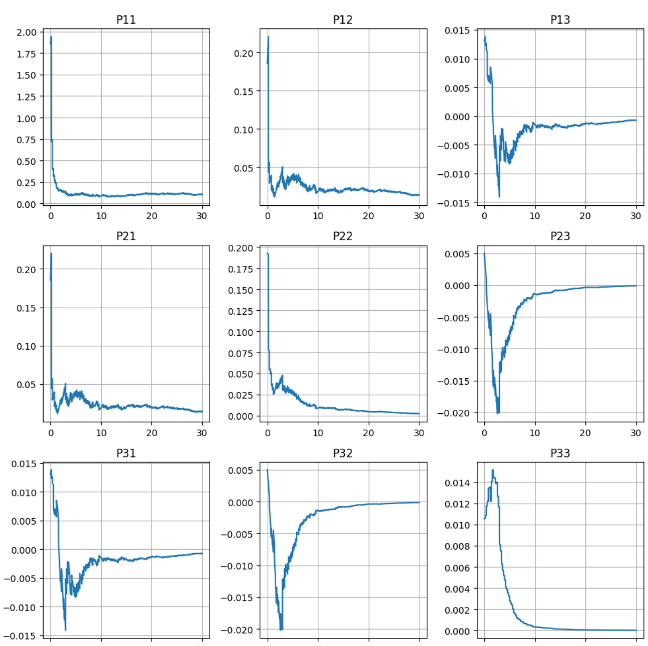

Fig. 10: Error Variance Average. Each plot shows a value of the $3x3$ error variance matrix over the 30-second interval.

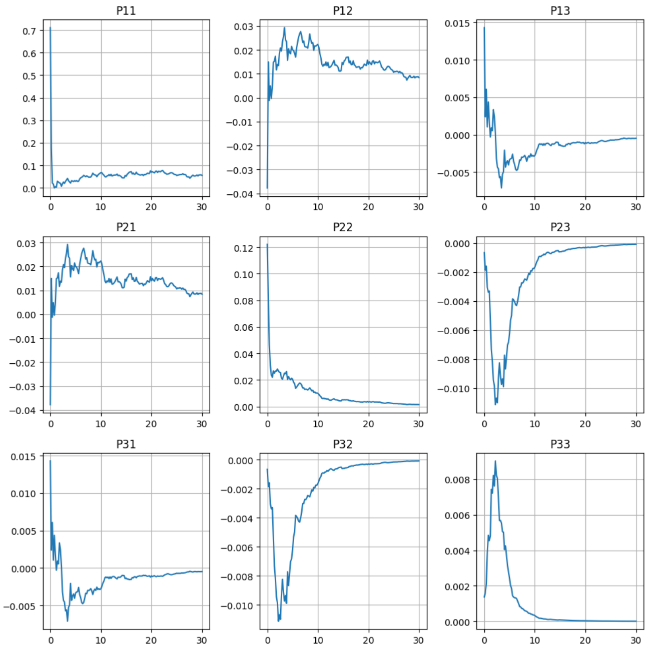

Fig. 11: Difference Between the covariance matrix computed in the Kalman filter algorithm and the Error variance average. Only the last realization of the covariance matrix is shown.Each plot shows a value of the $3x3$ error variance matrix over the 30-second interval.

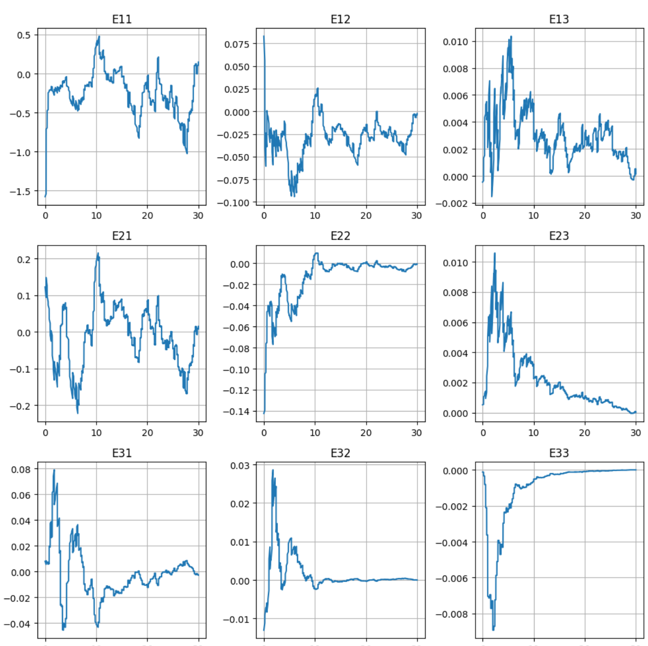

Fig. 12: Orthogonality of Error in Estimates and Estimate. Each plot shows a value of the $3x3$ error variance matrix over the 30-second interval.