MAE271A Zhijie Wang
Project UID: 304946569

Calibration of an Accelerometer using GPS Measurement

Zhijie Wang

Abstract

In this project, Kalman filter algorithm is implemented to improve the accuracy of an accelerometer with GPS measurement. Monte Carlo experiment is conducted on the mean error and mean covariance. The mean error is within one sigma bound. Then I further analyze the orthogonality to prove the correctness of the Kalman filter.

Introduction

Kalman filter is an algorithm intake a series of measurement containing noise, calculating the joint probability distribution and output estimates of unknown variables that tend to be more accurate. In this project, we are given a vehicle accelerates in one dimension in an inertial frame. The position, velocity, and acceleration is measured by an accelerometer. The position and velocity are also measured by a GPS. The goal is to calibrate the accelerometer using the GPS measurement. I first derived the models for the dynamic system, accelerometer and GPS measurement. Then I designed the Kalman filter according to the lectures. Then multiple tests were conducted to verify the performance of Kalman filter.

True Model

True model is first derived with given acceleration, which is harmonic form:

$$a(t) = 10\sin(2\pi\omega t)\ \mathrm{m/s^2}$$

where $\omega = 0.1$ rad/sec.
The velocity $v(t)$ can be calculated by taking integral of acceleration:

$$v(t) = v_0 + \frac{A}{2\pi\omega}(1 - \cos(2\pi\omega t))$$

where $v_0 = 100m/s$ is the initial velocity
The position $p(t)$ can be calculated by taking integral of velocity

$$p(t) = p_0 + \left(v_0 + \frac{A}{2\pi\omega}\right)t - \frac{A}{4\pi^2\omega^2}(\sin(2\pi\omega t))$$

where $p_0 = 0$ is the initial position

Accelerometer Model

The acceleration is measured by an accelerometer with a sample rate of 200 Hz at sample time t_j, where the sample period $t_{j+1} - t_j = 0.2s$. The accelerometer variance $V = 0.0004\ \left(\frac{\mathrm{m}}{\mathrm{s}^2}\right)^2$. The accelerometer has a bias b_a with a priori statistics: $b_a \sim N\left(0,0.1\left(\frac{\mathrm{m}}{\mathrm{s}^2}\right)^2\right)$. The accelerometer a_c is given as

$$a_c(t_{j+1}) = a(t_j) + b_a + \omega(t_j)$$

Using Euler method, we can obtain velocity:

$$v_c(t_{j+1}) = v_C(t_j) + a_C(t_j)\Delta t$$

and position can be derived as:

$$p_c(t_{j+1}) = p_c(t_j) + v_C(t_j)\Delta t + a_C(t_j)\left(\frac{\Delta t^2}{2}\right)$$

The initial conditions are given as $v_c(0) = 100\frac{m}{s}$, and $p_c(0) = 0$.
The above equations can be visualized in the following figure:

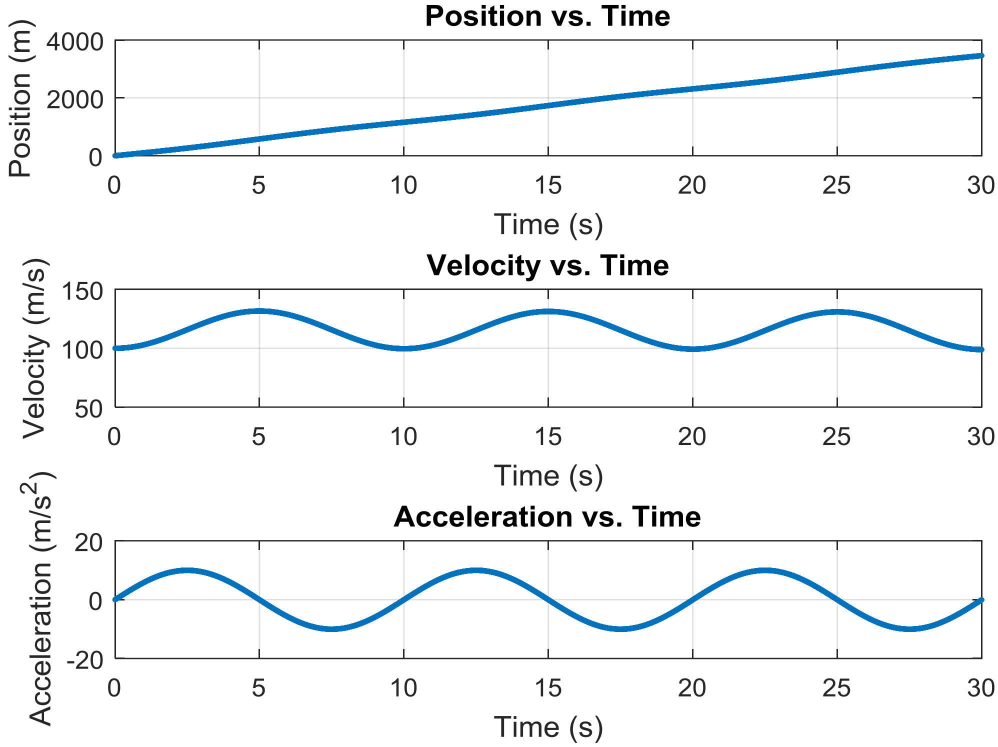

Figure 1. Accelerometer Model

Dynamic Model

Assumption is made that the actual acceleration is integrated by the same Euler integration formula as the accelerometer. So the estimated velocity and positon can be modeled as:

$$v_E(t_{j+1}) = v_E(t_j) + a(t_j)\Delta t$$

$$p_E(t_{j+1}) = p_E(t_j) + v_E(t_j)\Delta t + a(t_j)\left(\frac{\Delta t^2}{2}\right)$$

With the initial statistics $v(0)\sim N(v_0, M_0^v)$ and $p(0)\sim N(p_0, M_0^P)$
Then we subtract the actual model from the estimate mode to obtain the dynamic equations:

$$\begin{bmatrix}\delta p_{k+1}\\ \delta v_{k+1}\\ ba_{k+1}\end{bmatrix} = \begin{bmatrix}1 & \Delta t & \frac{\Delta t^2}{2}\\ 0 & 1 & \Delta t\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\delta p_k\\ \delta v_k\\ ba_k\end{bmatrix} - \begin{bmatrix}\frac{\Delta t^2}{2}\\ \Delta t\\ 0\end{bmatrix} w_k(t_k)$$

where the variables are given:

$$\delta p_E = p_e(k) - p_c(k) \sim N\left(\bar{p}_0, M_0^p\right)$$
$$\delta v_E = v_e(k) - v_c(k) \sim N(\bar{v}_0, M_0^v)$$
$$E[w(k)] = 0$$
$$E[w(k)w(t_i)^*] = W\delta_{ki}$$
$$b_0 \sim N(0, M_0^b)$$

Measurement

A GPS receiver is used to measure position and velocity in an inertial space. The measurements which are available at 5 Hz rate are:

$$z_i^p = p_i + \eta_{1i}$$
$$z_{2i}^v = v_i + \eta_{2i}$$

where x_i is the position and v_i is the velocity. Their a priori statistics are $x_0 \sim N(0\text{ m}, (10\text{ m})^2)$ and $v_0 \sim N\left(100\,\frac{\text{m}}{\text{s}}, \left(1\,\frac{\text{m}}{\text{s}}\right)^2\right)$. The additive measurement noises are assumed to be white noise sequences and independent of each other with statistics

$$\begin{bmatrix}\eta_{1i}\\ \eta_{2i}\end{bmatrix} \sim N\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}1\text{ m}^2 & 0\\ 0 & 4\left(\frac{\text{cm}}{\text{s}}\right)^2\end{bmatrix}\right)$$

To put the measurement equations in a convenient form to construct the Kalman filter. I subtract the computed accelerometer model from the GPS measurement and get:

$$\delta z_{1i} = \delta p_i + \eta_{1i}$$
$$\delta z_{2i} = \delta v_i + \eta_{2i}$$

Theory & Algorithm

Overview

The Kalman filter estimates the state of the system using a weighted average which is calculated from the covariance. The calculated weight average is the new state estimate which is relative accurate.

Initial state

Assume that the approximation can be made that $\delta p(t_i) = \delta pE(t_i)$ and $\delta v(t_i) = \delta vE(t_i)$ Posteriori state are defined as:

$$\widehat{\delta x}(t_i) = \begin{bmatrix}\widehat{\delta p}(t_i)\\ \hat{\delta}(t_i)\\ \hat{b}(t_i)\end{bmatrix} = \begin{bmatrix}\hat{p}(t_i) - p_c(t_i)\\ \hat{v}(t_i) - v_c(t_i)\\ \hat{b}(t_i)\end{bmatrix}$$

The state estimate is considered to be the estimate of the actual position and velocity which is what the measurement uses.

Kalman Filter Algorithm

The algorithm for Kalman filter is listed below, at each time step:

1. Calculate Kalman Gain at current stage
$$K_k = P_k H^T v_k^{-1}$$
2. Calculate the residue:
$$r_k = z_k - H_k \bar{x}_k$$
3. Calculate conditional mean at current stage
$$\hat{x}_k = \bar{x}_k + K_k(z_k - H_k \bar{x}_k)$$
4. Calculate covariance at current stage
$$P_k = M_k - M_k H_k^T (H_k M_k H_k^T + V_k)^{-1} H_k M_k$$
5. Propagated mean to next stage
$$\bar{x}_{k+1} = \Phi_k \hat{x}_k$$
6. Propagated covariance to next stage
$$M_{k+1} = \Phi_k P_k \Phi_k^T + \Gamma_k W_k \Gamma_k^T$$

Because the accelerator and GPS have different frequency. The above algorithm only works when the GPS measurement exists. If the GPS measurement doesn't exist, I used the following algorithm:

$$\bar{x}_{k+1} = \Phi_k \bar{x}_k$$
$$M_{k+1} = \Phi_k M_k \Phi_k^T + \Gamma_k W_k \Gamma_k^T$$

So an 'if statement' is implemented in the MATLAB code.

Results & Performance

The Performance of Kalman Filter

The performance of the Kalman filter is accessed using Monte Carlo simulation. The Monte Carlo simulation is to test the actual error variance. The formula is in the following form:

$$\delta x(t_i) = \begin{bmatrix} \delta p \\ \delta v \\ b \end{bmatrix} = \begin{bmatrix} p - p_c \\ v - v_c \\ b \end{bmatrix}$$

Where p, v, b are from the true model. The *a priori* estimation error is:

$$\bar{e}(t_j) = \delta x(t_j) - \overline{\delta x}(t_j) = \begin{bmatrix} p - \bar{p} \\ v - \bar{v} \\ b - \bar{b} \end{bmatrix}$$

Where $\overline{(.)}$ denotes the a priori estimate or estimation error propagated in the Kalman filter. The experiment for the single realization is plotted below:

The one sigma upper and lower bound for each plot is generated using the diagonal elements of covariance P.

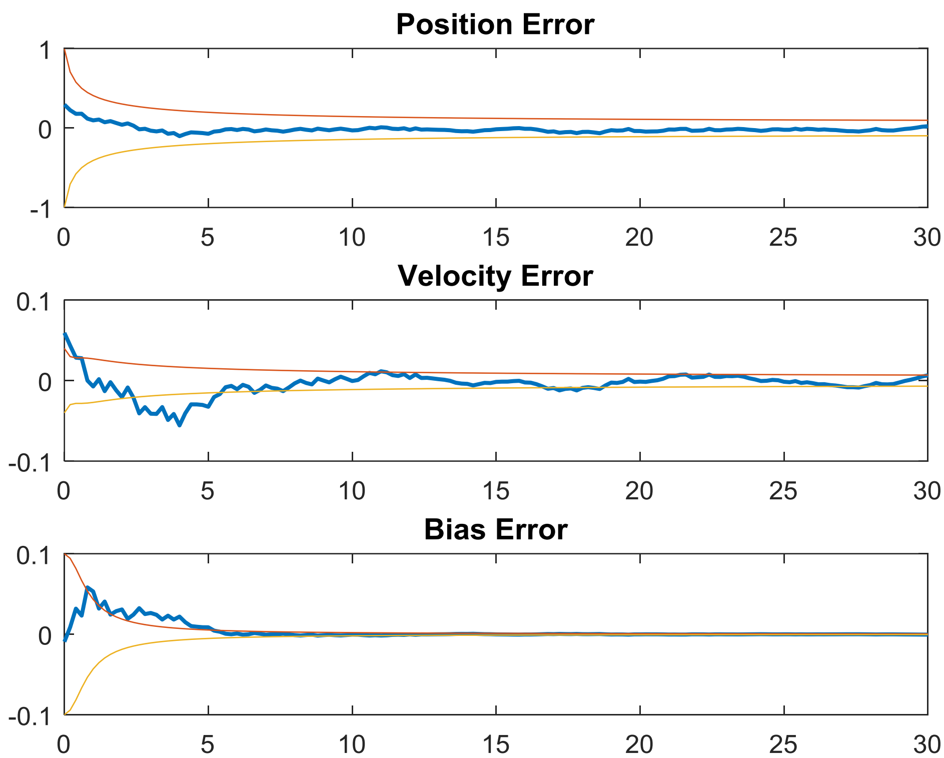

Figure 2. Error for a single realization with one sigma bound

From the plot above we can observe that the error for each dimension propagates to zero along time. It is also expected that most of the error lay within one sigma bound. The Kalman filter seems to perform well but more tests need to be conducted.

A Monte Carlo simulation is constructed to find the ensemble average over a set of realizations. The average of the error with N iteration is calculated as:

$$e^{avg}(t_i) = \frac{1}{N}\sum_{l=1}^{N} e^{l}(t_i)$$

It is expected that $e^{avg}(t_i)$ should close to 0 for all $t_i \in [0,30]$. The actual results of average error over 1000 realizations are plotted below:

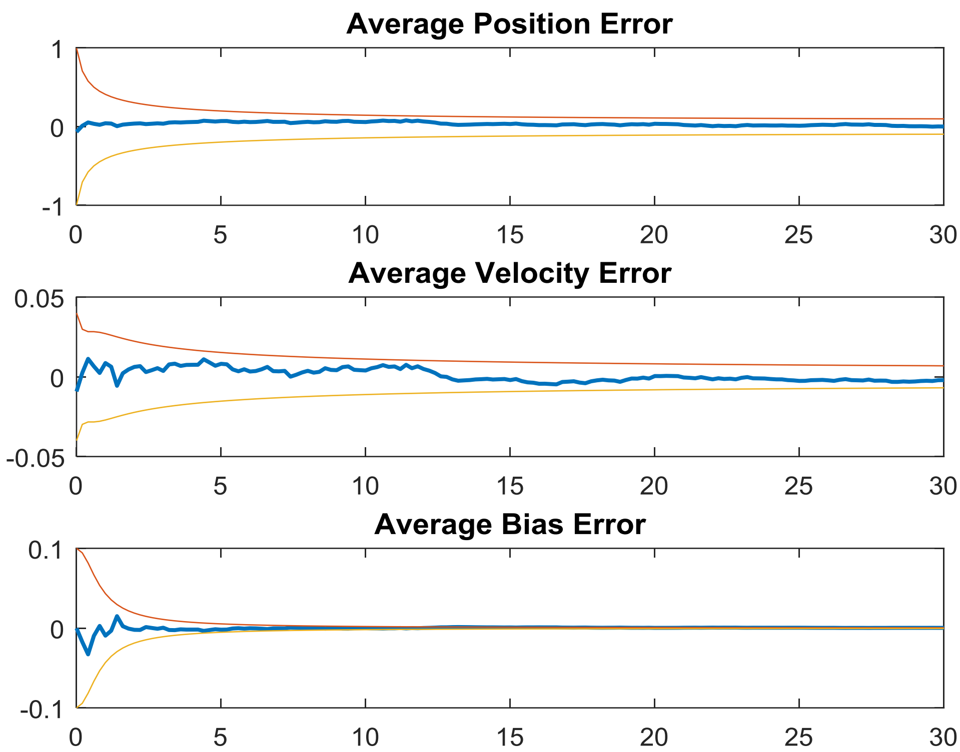

Figure 3. Error for multiple runs with one sigma bound (N = 1000)

From the plot we can see that all three dimension lies within one sigma bound. The error mean is close to zero, which is desired. The Monte Carlo experiment shows that the Kalman filter performs well.

Error Covariance Check

Next test is to show that the error variance obtained from the simulation and averaged over an ensemble of realization is close to the error variance used in the Kalman filter algorithm. The average of the error variance is calculated as:

$$P^{avg}(t_i) = \frac{1}{N-1}\sum_{l=1}^{N}[e^l(t_i) - e^{avg}(t_i)]\,[e^l(t_i) - e^{avg}(t_i)]^T$$

In the equation above, $N-1$ is used for an unbiased variance from small sample theory. The average of the error covariance P^{avg} is supposed to be close to $P(t_i)$ computed in the Kalman filter algorithm. Therefore it is expected that $P^{avg}(t_i) - P(t_i) = 0$ for all t. To test this, each element of $[P^{avg}(t_i) - P(t_i)]$ is plotted in Figure 4

Zhijie Wang
UID: 304946569

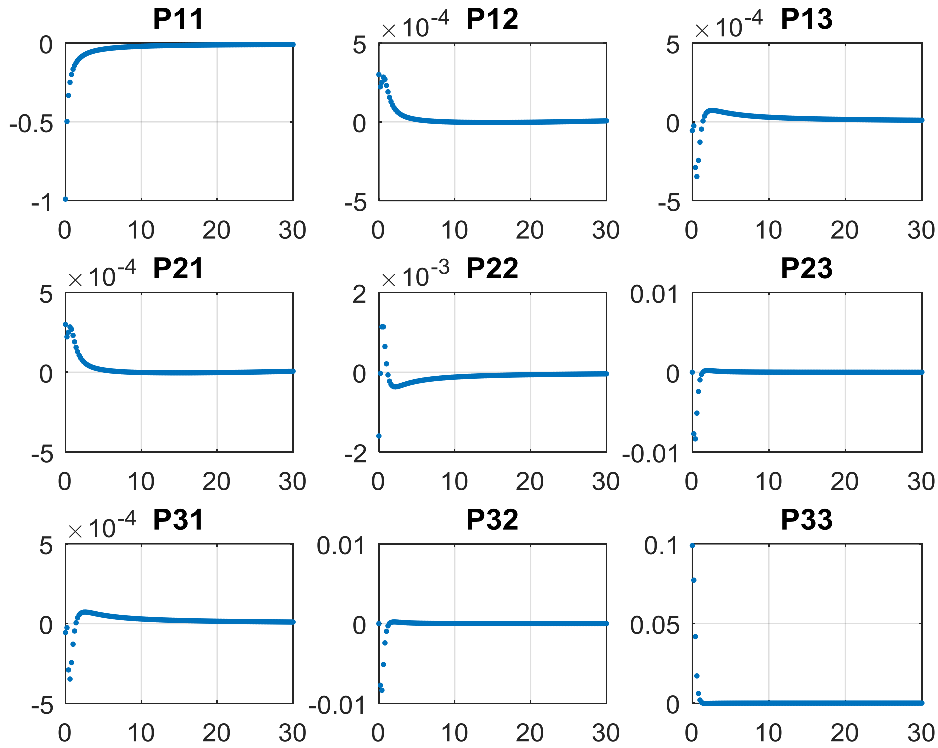

Figure 4. The deviation between average error covariance and each error covariance

From the plots above, we can observe that the value difference between $P^{avg}(t_i)$ and $P(t_i)$ is close to zero, which proves the correctness of the Kalman filter.

Orthogonality Check

The orthogonality of the error in estimates with the estimate is also check with equation:

$$\frac{1}{N}\sum_{l=1}^{N}[e^l(t_i) - e^{avg}(t_i)]\,\hat{x}(t_i)^T$$

Each element of the result matrix is plotted in next page:

Zhijie Wang
UID: 304946569

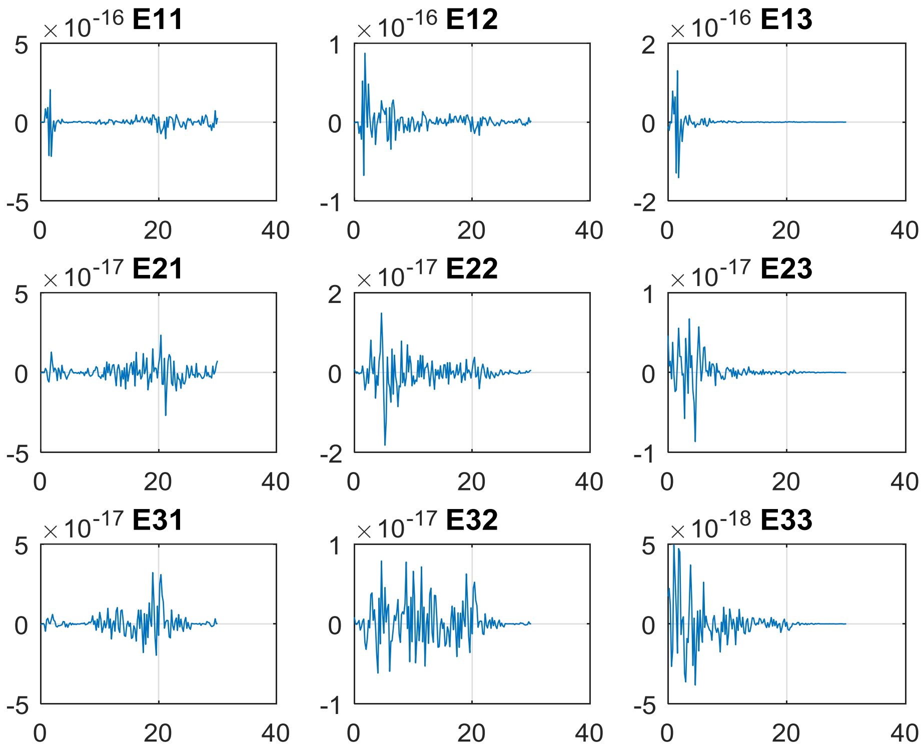

Figure 5. Orthogonality Check

From the plots above, we can see that the results are basically zero. The orthogonality is checked. Finally the independence of the residuals are to be checked. The residue for a realization l is:

$$r^l(t_i) = \begin{bmatrix} \delta z^{pl}(t_i) - \overline{\delta z^{pl}}(t_i) \\ \delta z^{vl}(t_i) - \overline{\delta z^{vl}}(t_i) \end{bmatrix}$$

The ensemble average for the correlation of the residue is:

$$\frac{1}{N_{avg}} \sum_{l=1}^{N_{avg}} r^l(t_i) r^l(t_m) = \begin{bmatrix} 0.00026480 & 0.00006487 \\ 0.00006487 & 0.00003597 \end{bmatrix} \approx 0$$

The norm of the above matrix is quite close to zero. So the orthogonality properties are checked.

Conclusion

In this project, a Kalman filter is implemented to calibrate the accelerometer with GPS measurement. Multiple tests were conducted on the average error, average covariance, and orthogonality. The result of tests all prove the success of the filter. From the result of the estimation, we can observe that the Kalman filter can improve the accuracy of the measurement.