ECE 8540

LAB 4 - KALMAN FILTER

November 3, 2021

Tim Driscoll
Clemson University
Department of Electrical and Computer Engineering
tdrisco@clemson.edu

1 Introduction

This lab report concerns the problem of applying a Kalman filter to a variety of measurements. The measurements are numerical data that has been recorded over time in order to provide samples to filter. There are two measurement sets, one that was recorded in one dimension and another recorded in two dimensions. In practice the Kalman filter would be used actively to filter the measurements in real time. The Kalman filter is used to more accurately determine the desired state variables. In practice there will be some level of noise that we want to remove. The Kalman filter allows us to weight the accuracy of our predictions and measurements to produce the best results. Previously, methods such as only using models or measurements to determine state have been implemented. There are also a number of other filtering techniques. When the Kalman filter was introduced it was highly regarded as one of the most effective and advanced filtering methods for tracking systems.

2 Methods

The Kalman filter follows a cycle of prediction and update. In this cycle the next state and state covariance is predicted. Then measurements are obtained from the sensor that is being actively used. After the measurements are obtained the Kalman gain is calculated. The Kalman gain represents the weights that are placed on both the predictions and measurements. The state and state covariance is then updated. This process is then looped over the next time step. These steps can be modeled in the follow series of equations:

1. Predict next state
$$X_{t,t-1} = \Phi X_{t-1,t-1} \tag{1}$$
2. Predict next state covariance
$$S_{t,t-1} = \Phi S_{t-1,t-1} \Phi^T + Q \tag{2}$$
3. Obtain measurement(s) Y_t
4. Calculate the Kalman gain (weights)
$$K_t = S_{t,t-1} M^T [M S_{t,t-1} M^T + R]^{-1} \tag{3}$$
5. Update state
$$X_{t,t} = X_{t,t-1} + K_t (Y_t - M X_{t,t-1}) \tag{4}$$
6. Update state covariance
$$S_{t,t} = [I - K_t M] S_{t,t-1} \tag{5}$$
7. Loop (t is incremented by Δt)

It is important to note that in this series of equations initial values need to be defined. These values include the sate at time 0, the state covariance at time 0, the dynamic noise covariance and the measurement noise covariance. It is the job of the designer to determine how these values should be defined based on available information. The designer must also define the state transition matrix, Φ, and the observation matrix, M. The transition matrix is determined by observing how each of the state variables contribute to each state transition equation. Similarly, the observation matrix is determined by observing how each of the state variables contribute to the observation equations.

3 1D Kalman Filter

For the one dimensional measurements that were being filtered for this lab a constant velocity model was used. The state variables were determined to be both position and velocity and were represented in the following matrix:

$$X_t = \begin{bmatrix} x_t \\ \dot{x}_t \end{bmatrix} \tag{6}$$

In matrix 6, x_t represents position and $\dot{x}_t$ represents the velocity. For this model at time 0 it was chosen that both the position and velocity would be set to 0. Since a constant velocity model was chosen the state transition equations could be modeled as follows.

$$x_{t+1} = x_t + T\dot{x}_t \tag{7}$$

$$\dot{x}_{t+1} = \dot{x}_t \tag{8}$$

These equations then allowed for the state transition matrix to be produced. This matrix was defined by matrix 9 below.

$$\Phi = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \tag{9}$$

For this example the value T was set to equal one. T was set equal to one because the time step for each iteration of new measurements was defined to be one second. It was then important to define the state estimate covariance, S_t. The state estimate covariance represents the level of uncertainty that there is in the state variables shown in matrix 6. Specifically, the diagonal of matrix 10 represents the uncertainty of the position and velocity estimates. The elements off of the diagonal represent the covariances of the position and velocity estimates.

$$S_t = \begin{bmatrix} \sigma_x^2 & \sigma_{x,\dot{x}} \\ \sigma_{x,\dot{x}} & \sigma_{\dot{x}}^2 \end{bmatrix} \tag{10}$$

For this model at time 0 the state covariance matrix was defined to be the identity matrix. This placed an initial uncertainty on the state variables, but not their covariances. Following this step the observation equations and variables were defined. For this model it was assumed that the position was being sensed. The observation variable was defined in the following matrix:

$$Y_t = \begin{bmatrix} \tilde{x}_t \end{bmatrix} \tag{11}$$

The observation equation was also simply defined as the following:

$$\tilde{x}_t = x_t \tag{12}$$

These definitions allowed for the observation matrix, M to be defined by the 1x2 matrix below.

$$M = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{13}$$

The final two variables that needed to be initialized were the dynamic noise covariance and the the measurement noise covariance, represented by Q and R respectively. These matrices would directly impact how the filtering would behave. Depending on the ratio between the two noises the filter would either more closely follow the measurements or the state transition equations. In this lab the noises were tuned to adjust the behavior of the filter and observe the results. The generic form of both the Q and R matrix used for this 1D model are as follows:

$$Q = \begin{bmatrix} 0 & 0 \\ 0 & \sigma_a^2 \end{bmatrix} \tag{14}$$

$$R = [\sigma_n^2] \tag{15}$$

In matrix 15 the singular value represents the variance of the measurement noise of position. There is no dynamic noise on the position portion of the state transition equations. Thus matrix 14 is only composed of the variance for the velocity portion of the dynamic noise. Both the dynamic noise and measurement noise matrices were not explicitly defined in this section because they can be shown in an embedded form in the Q and R matrices.

4 2D Kalman Filter

For the second set of sample measurements a 2D constant velocity model was used. The 2D constant velocity model is an extension of the 1D constant velocity model with additional state variables. The state variables were determined to be both position and velocity and were represented in the following matrix:

$$X_t = \begin{bmatrix} x_t \\ \dot{x}_t \\ y_t \\ \dot{y}_t \end{bmatrix} \tag{16}$$

In matrix 16 x_t and y_t represent position and both $\dot{x}_t$ and $\dot{y}_t$ represent the velocities. For this model at time 0 it was chosen that both the position values would be set to the initial measurement and velocity values would be set to 0. Since a constant velocity model was chosen the state transition equations could be modeled the same as previously shown with two additional equations.

$$\begin{aligned} x_{t+1} &= x_t + T\dot{x}_t \\ \dot{x}_{t+1} &= \dot{x}_t \\ y_{t+1} &= y_t + T\dot{y}_t \\ \dot{y}_{t+1} &= \dot{y}_t \end{aligned} \tag{17}$$

These equations then allowed for the state transition matrix to be produced. This matrix was defined by matrix 18 below.

$$\Phi = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{18}$$

For this example the value T was also set to equal one. The state covariance matrix was again defined for the 2D model. The matrix is now size 4x4 because there are four state variables and for state transition equations. The state covariance matrix still follows the same properties that were mentioned above.

$$S_t = \begin{bmatrix} \sigma^2_{x_t} & \sigma_{x_t,y_t} & \sigma_{x_t,\dot{x}_t} & \sigma_{x_t,\dot{y}_t} \\ \sigma_{x_t,y_t} & \sigma^2_{y_t} & \sigma_{y_t,\dot{x}_t} & \sigma_{y_t,\dot{y}_t} \\ \sigma_{x_t,\dot{x}_t} & \sigma_{y_t,\dot{x}_t} & \sigma^2_{\dot{x}_t} & \sigma_{\dot{x}_t,\dot{y}_t} \\ \sigma_{x_t,\dot{y}_t} & \sigma_{y_t,\dot{y}_t} & \sigma_{\dot{x}_t,\dot{y}_t} & \sigma^2_{\dot{y}_t} \end{bmatrix} \tag{19}$$

For this model at time 0 the state covariance matrix was again defined to be the identity matrix. This placed an initial uncertainty on the state variables, but not their covariances. Following this step the observation equations and variables were defined. For this model is was assumed that the position was being sensed in both the x and y direction. The observation variables were defined in the following matrix:

$$Y_t = \begin{bmatrix} \tilde{x}_t \\ \tilde{y}_t \end{bmatrix} \tag{20}$$

The observation equations were also simply defined as the following:

$$\begin{aligned} \tilde{x}_t &= x_t \\ \tilde{y}_t &= y_t \end{aligned} \tag{21}$$

The equations are defined like this because the measurement data being received is position data directly. These definitions allowed for the observation matrix, M to be defined by the 2x4 matrix below.

$$M = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{22}$$

The dynamic noise and measurement covariances were again the final two matrices defined in the setup of the 2D Kalman filter. Both the noises were again tuned to adjust the behavior of the filter and determine what the best ratio was. The generic form of both the Q and R matrix used for this 2D model are as follows:

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma^2_{a_1} & \sigma_{a_1,a_2} \\ 0 & 0 & \sigma_{a_1,a_2} & \sigma^2_{a_2} \end{bmatrix} \tag{23}$$

$$R = \begin{bmatrix} \sigma^2_{n_1} & \sigma_{n_1,n_2} \\ \sigma_{n_1,n_2} & \sigma^2_{n_2} \end{bmatrix} \tag{24}$$

Table 1: Summary of three noise ratios observed for 1D measurements.

Ratio Case	Q (σ_a^2)	R (σ_n^2)
1	10	1
2	10^{-12}	10^{7}
3	10^{-7}	1

In matrix 24 the values on the diagonal represent the variances of the measurement noise of the positions, whereas off the diagonal represents their covariance. There is again no dynamic noise on the position portion of the state transition equations. Thus matrix 23 is only composed of the variances and covariances for the velocity portion of the dynamic noise. Both the dynamic noise and measurement noise matrices were not explicitly defined in this section because they can be shown in an embedded form in both the Q and R matrices respectively.

4.1 MATLAB Implementation

Once it was determined how the matrices were defined for each data set equations 1-5 were implemented in software. The 2020a version of MATLAB on a Windows operating system was used to implement the code. All the initial matrices were constructed in the code and a for loop was used to iterate over the data for each time step. It was assumed that a new measurement would be read every second, thus ΔT was defined to be 1. The loop was ran over the entire length of the measurement data. The updated state was calculated and stored in a separate matrix at the end of each loop. This saved data was then used to plot the results of the filtering for each of the measurement sets.

5 Results

5.1 1D Kalman Filter

The 1D Kalman filter was tuned and observed at three different ratios of dynamic noise and measurement noise. The tuning of this filter was done by holding the measurement noise as a constant value and adjusting the dynamic noise to alter the ratio between the two. The three different ratios represent the three different general cases that can be observed in the estimated data outputted by the Kalman filter. A summary of the variances used for each case can be observed in table 1. Note that both the measurement variance and the dynamic variance were adjusted in case two so the desired results were achieved. The first case is when the measurement noise is smaller then the dynamic noise. This ratio will add more weight to the measurements and the estimated output will more closely resemble the measurements. If the difference between the two noises in this case is large enough the estimated output will almost completely match the measurement data. As shown by figure 1 the filtered estimate almost completely overlaps the the measurement data. The measurement data is displayed in red and is barely visible behind the black filtered estimate showing how closely they match.

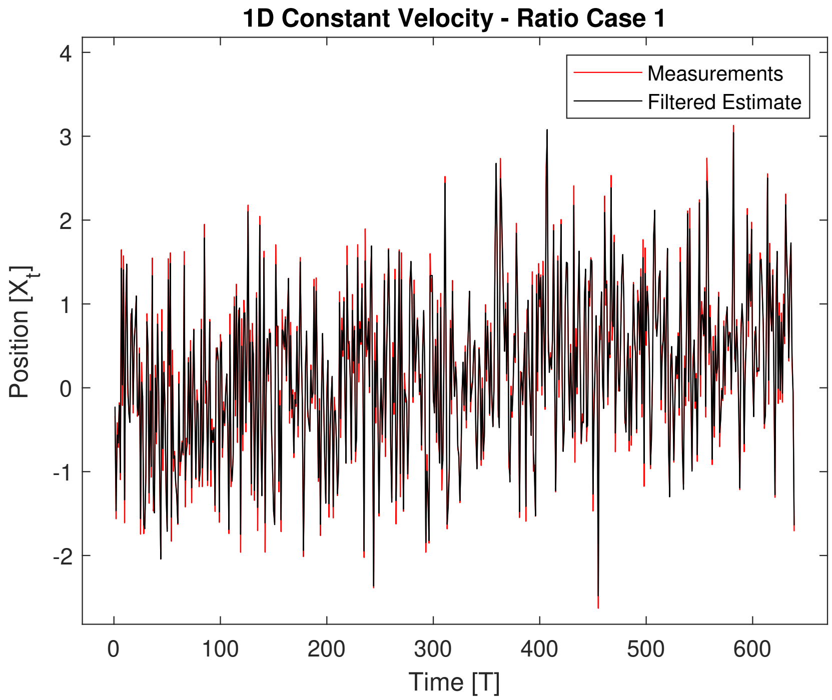

Figure 1: Raw measurement data and filtered estimate when $(\sigma_a^2) = 10$ and $(\sigma_n^2) = 1$.

If the dynamic noise variance was increased by another power of 10 the the filtered estimate would appear to perfectly match the measurements. The second ratio case was when the dynamic noise was much smaller then the measurement noise. With this ratio the behavior of the filtered estimate should more closely match the values produced by equation 7. Since this model is a constant velocity model the filtered estimate should produce a straight line. When observing figure 2 the filtered estimate almost represents a straight line. The line appears to have a slightly positive slope due to the fact that there is still a small amount of weight on the measurements. If the dynamic noise and measurement noise variances were further exaggerated a straight line would be produced by the filtered estimate. The final case was more of a balanced ratio between the dynamic noise and measurement noise variance. The balanced case allowed for a more equal weighting to be placed on the measurements and the state equation prediction. The ratio that was chose places slightly more weight on the prediction to produce an estimate more closely resembling a straight line then the direct measurements. This final case can be observed in figure 3.

5.2 2D Kalman Filter

For the 2D Kalman filter sample measurements from ultra-wideband (UWB) tracking system was used. Ultra-wideband is a wireless communication protocol that can use emitted pulses to sense distance between two transmitters. The 2D measurements produced by the UWB tracking were filtered using a constant velocity model. A series of different ratios of dynamic noise variance and measurement noise variance were used to determine which would produce

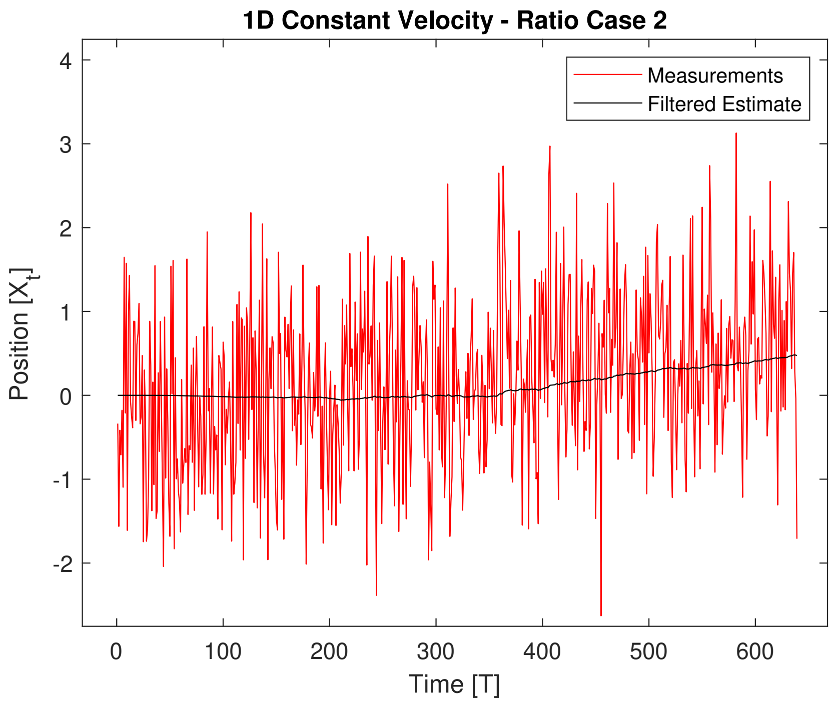

Figure 2: Raw measurement data and filtered estimate when $(\sigma_a^2) = 10^{-12}$ and $(\sigma_n^2) = 10^7$.

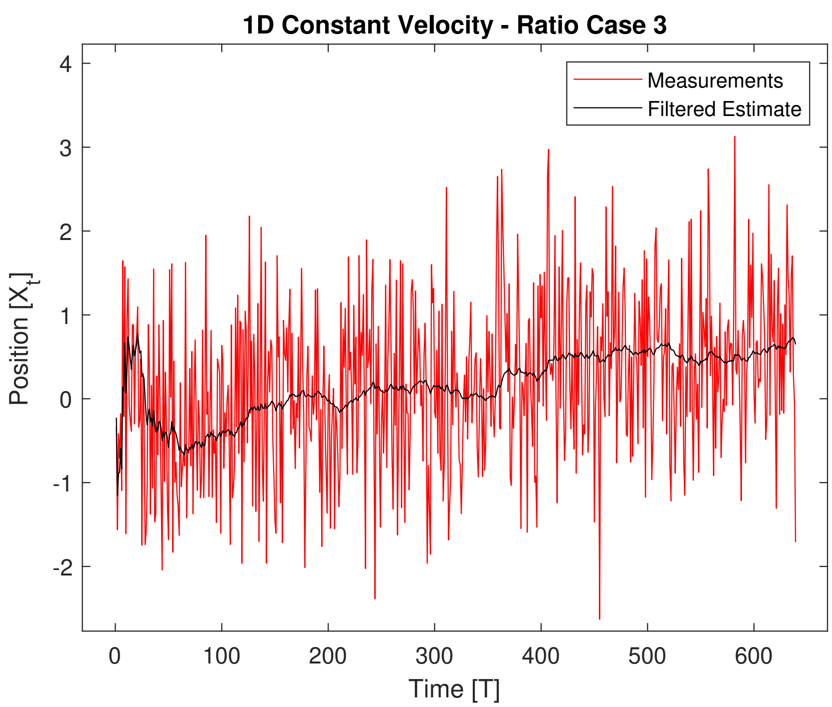

Figure 3: Raw measurement data and filtered estimate when $(\sigma_a^2) = 10^{-6}$ and $(\sigma_n^2) = 1$.

Table 2: Summary of dynamic noise covariance for 2D measurements.

Q	σ_{a1}^2	$\sigma_{a1,a2}$	$\sigma_{a1,a2}$	σ_{a2}^2
	0.01	0.0001	0.0001	0.01

Table 3: Summary of measurement noise covariance for 2D measurements.

R	σ_{n1}^2	$\sigma_{n1,n2}$	$\sigma_{n1,n2}$	σ_{n2}^2
	10	0.0001	0.0001	10

the best filtered estimates. The word best in this scenario is very opinionated because there is no ground truth for the measurement data. There is really no way to determine if the filtered estimate is the best estimate without having further information on the problem and measurement data. The dynamic noise covariance matrix was defined by the values shown in table 2. It was decided that equal variance be placed on both velocity portions of the dynamic noise. There was also a very small amount of covariance placed on the velocity portion of the dynamic noise. The filtered results were observed with and without this covariance and there were only slight differences in the estimated state. The measurement noise covariance matrix was defined by the values shown in table 3. A similar approach was used to define the measurement noise covariance matrix. An equal amount of noise was placed on the variance of both the x and y position measurements, and the covariance was defined to be a very small number. The ratio of dynamic noise and measurement noise used for this sample data most closely represents case three from the section above. The noise between the two were close, but there was a greater weight placed on the predicted state equations. Both figures 4 and 5 represent the measurement data and filtered estimate for X and Y position respectively. The filtered estimate used the values for the dynamic noise and measurement noise covariance defined in tables 2 and 3. The more closely balanced ratio allowed for the filtered estimate to follow the general trend of the measurements while smoothing out some of the sharp jumps that were interpreted as noise. For example in figure 4 the general sinusoidal trend of the measurements are followed, but the filtered estimate has a smoother output. Similar results can be observed in figure 5. A relatively straight line is tracked in the Y position. Given the weights, where there are large jumps in the measurements the filtered estimate also gets pulled down, but to a lesser degree.

6 Conclusion

This lab clearly showed how the Kalman filter uses measurements in conjunction with state predictions to produce estimates. The Kalman filter allows for the designer to place varying weights on both the measurements and predictions to gain the most accurate state estimates and handle noise. These varying weights can be achieved by adjusting the ratio between dynamic and measurement noise. If the measurement noise is held constant and the dynamic noise is increased an increasing amount of weight will be placed on the measurements. The exact opposite is true if the measurement noise is held constant and the dynamic noise is

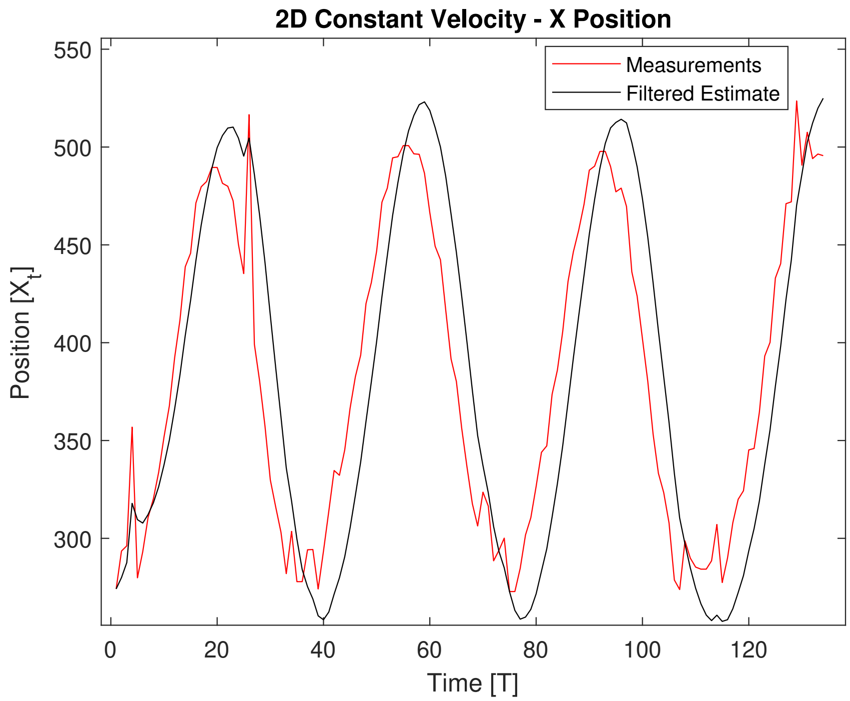

Figure 4: Raw UWB measurement data and filtered estimate of X position.

Figure 5: Raw UWB measurement data and filtered estimate of Y position.

decreased. In the lab the results behaved according to the relationship between dynamic and measurement noise. Although this is true it is hard to determine how accurate the filter estimates actually were. Without a ground truth there is now way to compare a filtered estimate. This is the main limitation of the Kalman filter implemented in this lab. In order to accurately tune the filter more information on the measurements would be needed. This could include information such as if the sensor has any known covariance between its measurements. In order to effectively use the Kalman filter a strong understanding of the problem, sensors and expected behavior is important when trying to optimize the estimate.