Journal of Engineering and Development, Vol. ١٧, No.١, Mar. ٢٠١٣, ISSN ١٨١٣- ٧٨٢٢

Solving Linear Systems Problems Using Discrete Kalman Filter

Yasin Yousif Mohammed
Al-Mustansirya University, College of Engineering,
Electrical Engineering Department

Abstract

This paper deals with the software implementation of discrete Kalman filter. It introduces discrete Kalman filter algorithm and then looks at the use of this filter to solve linear systems problems such as a vehicle navigation problem. By using discrete Kalman filter, one can measure the position every T seconds given that the acceleration is changing.
The software is implemented using the programming language of MATLAB Package and then it is tested under different conditions and circumstances. Finally, from testing results many points are then concluded.

Keywords: *Kalman Filter, Discrete Kalman Filter, Linear systems Problems, Stochastic Estimation, and Vehicle Navigation Problem.*

الخلاصة

أن هذا البحث يتناول بناء برنامج حاسوبي لنمذجة عمل مرشح (Kalman) المتقطع مع الزمن.
يقوم البحث في البداية بتوضيح خوارزمية مرشح (Kalman) المتقطع مع الزمن ومن ثم يطرح البحث كيفية أستخدام المرشح لحل مشاكل المنظومات الخطية مثل مشكلة قيادة مركبة.
ومن ثم فأن أي شخص يستطيع بأستخدام مرشح (Kalman) المتقطع مع الزمن قياس أزاحة المركبة عند كل (T) ثانية في حالة تغير تعجيلها.
تم بناء البرنامج بأستخدام اللغة البرمجية لبرنامج (MATLAB) وبعدها تم أختبار المنظومة التي تم بناءها في حالات و ظروف مختلفة ومن نتائج الأختبار تم بعد ذلك أستنتاج عدة نقاط.

١. Introduction

The Kalman filter is essentially a set of mathematical equations that implement a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated error covariance (when some presumed conditions are met). Since the time of its introduction, the Kalman filter has been the subject of extensive research and application, particularly in the area of autonomous or assisted navigation. This is likely due in large part to advances in digital computing that made the use of the filter practical, but also to the relative simplicity and robust nature of the filter itself. Rarely do the conditions necessary for optimality actually exist, and yet the filter apparently works well for many applications in spite of this situation. The Kalman filter has been used for tracking in interactive computer graphics, for motion prediction, and for multi-sensor (inertial-acoustic) fusion [١].

٢. The Discrete Kalman Filter Algorithm

The Discrete Kalman filter estimates a process by using a form of feedback control: the filter estimates the process state at some time and then obtains feedback in the form of (noisy) measurements. As such, the equations for the Kalman filter fall into two groups: time update equations and measurement update equations. The time update equations are responsible for projecting forward (in time) the current state and error covariance estimates to obtain the a priori estimates for the next time step. The measurement update equations are responsible for the feedback; i.e. for incorporating a new measurement into the a priori estimate to obtain an improved a posteriori estimate [٢, ٣].

The time update equations can also be thought of as predictor equations, while the measurement update equations can be thought of as corrector equations. Indeed the final estimation algorithm resembles that of a predictor-corrector algorithm for solving numerical problems as shown in Figure ١.

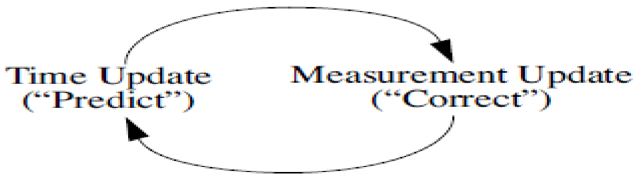

Figure ١ The ongoing discrete Kalman filter cycle.

The specific equations for the time and measurement updates are presented in Table ١ and Table ٢.

Table ١ Discrete Kalman filter time update equations.

$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k$	(١)
$P_k^- = AP_{k-1}A^T + Q$	(٢)

Where $\hat{x}_k^-$ and P_k^- are state and error covariance estimates and $\hat{x}_{k-1}$ and P_{k-1} are state and error covariance at time step k-١. The matrix A relates the state at the previous time step k-١ to the state at the current step k, in the absence of either a driving function or process noise. It is clear that in practice A might change with each time step, but here it is assumed constant. The matrix B relates the optional control input u to the state x. The matrix Q is the process noise covariance [٤, ٥].

It is clear that the time update equations in Table ١ project the state and covariance estimates forward from time step k-١ to step k.

Table ٢ Discrete Kalman filter measurement update equations.

$K_k = P_k^- C^T (CP_k^- C^T + R)^{-1}$	(٣)
$\hat{x}_k = \hat{x}_k^- + K_k(z_k - C\hat{x}_k^-)$	(٤)
$P_k = (I - K_k C)P_k^-$	(٥)

Where K_k is the Kalman gain. The matrix C relates the state to the measurement z_k. The matrix R is the measurement noise covariance.
Figure ٢ below offers a complete picture of the operation of the filter, combining the high-level diagram of Figure ١ with the equations from Table ١ and Table ٢ [٦, ٧].

٣. Vehicle Navigation Problem

Suppose one want to model a vehicle going in a straight line. Then it is clear that the state consists of the vehicle position p and velocity v. The input u is the commanded acceleration and the output z is the measured position. Then the aim of this problem is the ability to change the acceleration and measure the position every T seconds [٨, ٩].
In this case, elementary laws of physics say that the velocity v will be governed by the following equation:

$$v_k = v_{k-1} + Tu_k \qquad (٦)$$

That is, the velocity one time-step from now (T seconds from now) will be equal to the present velocity plus the commanded acceleration multiplied by T. But the previous equation does not give a precise value for $v_{k-١}$. Instead, the velocity will be perturbed by noise due to gusts of wind, and other unfortunate realities.

The velocity noise is a random variable that changes with time. So a more realistic equation for v would be:

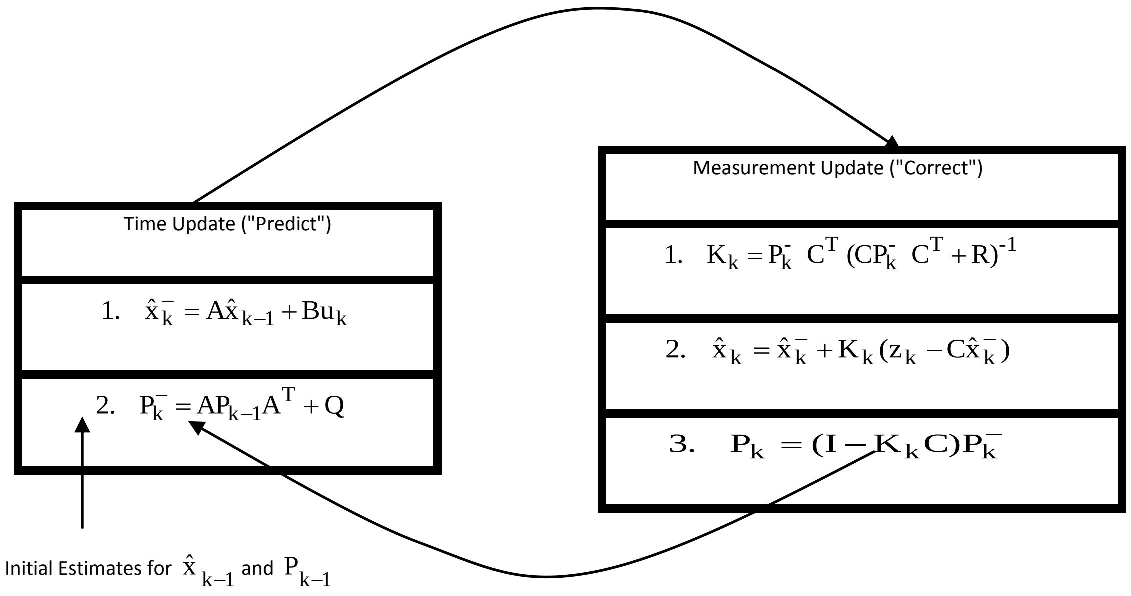
Time Update ("Predict")
1. $\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k$
2. $P_k^- = AP_{k-1}A^T + Q$
Measurement Update ("Correct")
1. $K_k = P_k^- C^T (CP_k^- C^T + R)^{-1}$
2. $\hat{x}_k = \hat{x}_k^- + K_k(z_k - C\hat{x}_k^-)$
3. $P_k = (I - K_k C)P_k^-$
Initial Estimates for $\hat{x}_{k-1}$ and P_{k-1}

Figure ٢ A complete picture of the operation of the Kalman filter, combining the high level diagram of Figure ١ with the equations from Table ١ and Table ٢.

$$v_k = v_{k-1} + Tu_k + \tilde{v}_{k-1} \qquad (٧)$$

where $\tilde{v}_{k-1}$ is the velocity noise. A similar equation can be derived for the position p:

$$p_k = p_{k-1} + Tv_{k-1} + \frac{1}{2}T^2 u_k + \tilde{p}_{k-1} \qquad (٨)$$

where $\tilde{p}_{k-1}$ is the position noise. Now a state vector x can be defined which consists of position and velocity:

$$x_k = \begin{bmatrix} p_k \\ v_k \end{bmatrix} \qquad (٩)$$

Finally, knowing that the measured output is equal to the position, one can write linear system equations as follows:

$$x_k = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} u_k + w_{k-1}$$

$$z_k = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + v_k \qquad (١٠)$$

v_k is the measurement noise due to such things as instrumentation errors [٨, ٩].

٤. Solving Vehicle Navigation Problem Using Discrete Kalman Filter

Suppose a linear system model is given as described previously and one want to use the available measurements z to estimate the state of the system x. An estimator is required that gives an accurate estimate of the true state even though one cannot directly measure it. Two obvious requirements come to mind.

١. The expected value of the estimate should be equal to the expected value of the state.

٢. The estimator should be having the smallest possible error variance.

The Kalman filter is the estimator that satisfies these two criteria. But the Kalman filter solution does not apply unless certain assumptions can be satisfied about the noise that affects system under consideration. From the given system model, w and v are defined as the process noise and the measurement noise respectively. It is assumed that the average value of w is zero and the average value of v is zero. Further it is assumed that no correlation exists between w and v. That is, at any time k, w_k, and v_k are independent random variables [١٠].

Then the noise covariance matrices S_w and S_v are defined as:

Process noise covariance

$$S_w = E(w_k w_k^T) \qquad (١١)$$

Measurement noise covariance

$$S_v = E(v_k v_k^T) \qquad (١٢)$$

where w^T and v^T indicate the transpose of the w and v random noise vectors, and $E(\cdot)$ means the expected value [١٠].

Then the ongoing discrete Kalman filter cycle shown in Figure ١ and Figure ٢ can be used to solve vehicle navigation problem as shown in the following flowchart.

٥. Software Testing

The implemented software is tested in this section to observe the effect of the following parameters:

١. Varying step size for different durations.

٢. Varying position measurement noise.

٣. Varying acceleration noise.

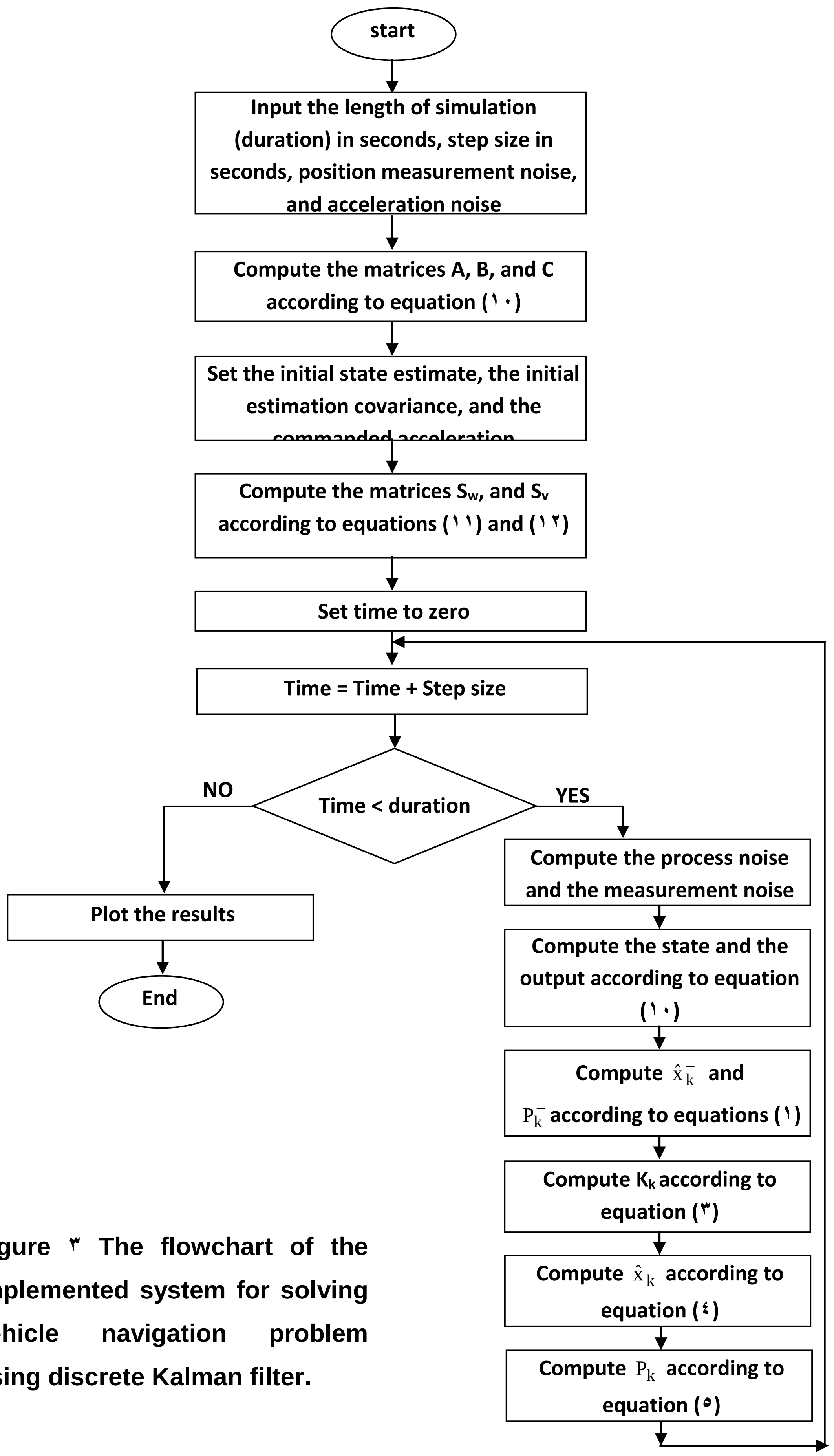
start
Input the length of simulation (duration) in seconds, step size in seconds, position measurement noise, and acceleration noise
Compute the matrices A, B, and C according to equation (١٠)
Set the initial state estimate, the initial estimation covariance, and the commanded acceleration
Compute the matrices Sw, and Sv according to equations (١١) and (١٢)
Set time to zero
Time = Time + Step size
Time < duration
NO
YES
Plot the results
End
Compute the process noise and the measurement noise
Compute the state and the output according to equation (١٠)
Compute x̂k⁻ and Pk⁻ according to equations (١)
Compute Kk according to equation (٣)
Compute x̂k according to equation (٤)
Compute Pk according to equation (٥)

Figure ٣ The flowchart of the implemented system for solving vehicle navigation problem using discrete Kalman filter.

٥.١ The Effect of Varying Step Size for Different Durations

Figure ٤ to Figure ٩ presents the results of executing the implemented software under the following conditions.

١- Duration = ٣٠ seconds, step size =١ seconds (Figure ٤), ٠,٥ seconds (Figure ٥), ٠,١ seconds (Figures ٦), position measurement noise =١٠ ft, and acceleration noise ٠,٢ ft/sec٢. The commanded acceleration in this case is ١ ft/sec٢.

٢- Duration = ٦٠ seconds, step size =١ seconds (Figure ٧), ٠,٥ seconds (Figure ٨), ٠,١ seconds (Figures ٩), position measurement noise =١٠ ft, and acceleration noise ٠,٢ ft/sec٢. The commanded acceleration in this case is ١ ft/sec٢.

From the above figures, the following points are concluded:

١. By decreasing the step size, it is clear that true and estimated vehicle positions become so close from each other.

٢. By decreasing the step size, it is clear that position estimation error will be improved.

٣. By decreasing the step size, it is clear that true and estimated vehicle velocities become so close from each other.

٤. By decreasing the step size, it is clear that velocity estimation error will be improved.

٥.٢ The Effect of Varying Position Measurement Noise

Figure ١٠ to Figure ١٢ presents the results of executing the implemented software under the following conditions.

١. Duration = ٦٠ seconds, step size = ٠,١, position measurement noise = ٥ ft, and acceleration noise ٠,٢ ft/sec٢. The commanded acceleration in this case is ١ ft/sec٢ (Figure ١٠).

٢. Duration = ٦٠ seconds, step size = ٠,١, position measurement noise =١٠ ft, and acceleration noise ٠,٢ ft/sec٢. The commanded acceleration in this case is ١ ft/sec٢ (Figure ١١).

٣. Duration = ٦٠ seconds, step size = ٠,١, position measurement noise =٢٠ ft, and acceleration noise ٠,٢ ft/sec٢. The commanded acceleration in this case is ١ ft/sec٢ (Figure ١٢).

From the above figures, the following points are concluded:

١. By increasing position measurement noise, it is clear that measured vehicle position will be degraded.

٢. By increasing position measurement noise, it is clear that position measurement error will be increased.

٥.٣ The Effect of Varying Acceleration Noise

Figure ١٣ to Figure ١٥ presents the results of executing the implemented software under the following conditions.

١. Duration = ٦٠ seconds, step size = ٠,١, position measurement noise =١٠ ft, and acceleration noise ٠,٢ ft/sec^{٢}. The commanded acceleration in this case is ١ ft/sec^{٢} (Figure ١٣).

٢. Duration = ٦٠ seconds, step size = ٠,١, position measurement noise =١٠ ft, and acceleration noise ٠,٥ ft/sec^{٢}. The commanded acceleration in this case is ١ ft/sec^{٢} (Figure ١٤).

٣. Duration = ٦٠ seconds, step size = ٠,١, position measurement noise =١٠ ft, and acceleration noise ١ ft/sec^{٢}. The commanded acceleration in this case is ١ ft/sec^{٢} (Figure ١٥).

From the above figures, the following points are concluded:

١. By increasing acceleration noise, it is clear that true and estimated vehicle positions become so far from each other.

٢. By increasing acceleration noise, it is clear that position estimation error will be increased.

٣. By increasing acceleration noise, it is clear that true and estimated vehicle velocities become so far from each other.

٤. By increasing acceleration noise, it is clear that velocity estimation error will be increased.

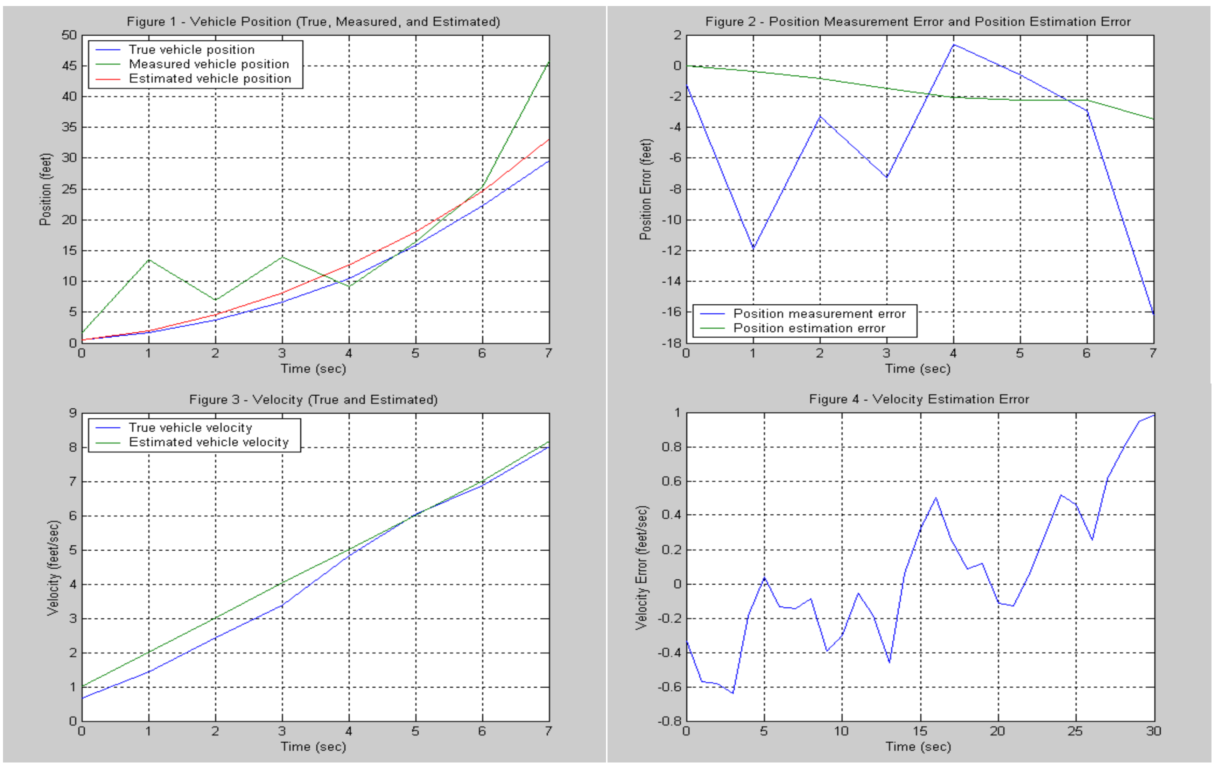

Figure ٤ Execution results for Duration = ٣٠ seconds, step size = ١ seconds, position measurement noise =١٠ ft, and acceleration noise ٠,٢ ft/sec^{٢}.

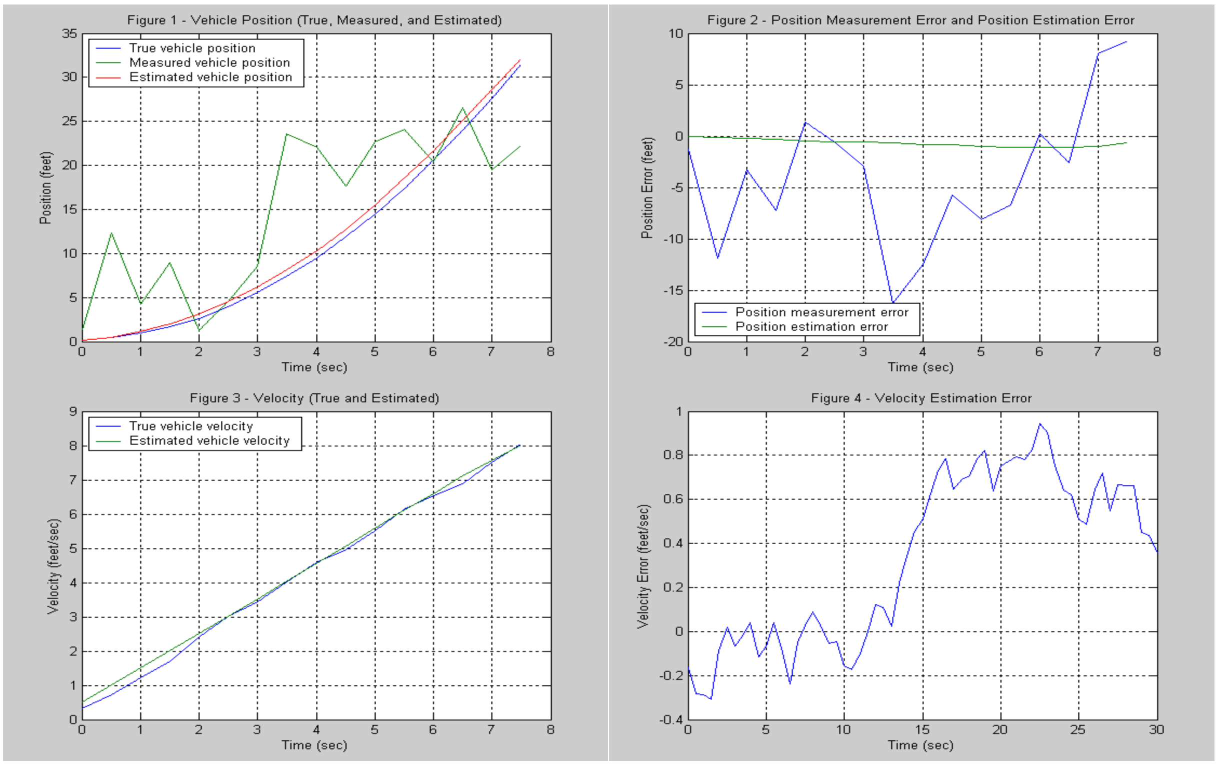

Figure ٥ Execution results for Duration = ٣٠ seconds, step size = ٠,٥ seconds, position measurement noise =١٠ ft, and acceleration noise ٠,٢ ft/sec٢.

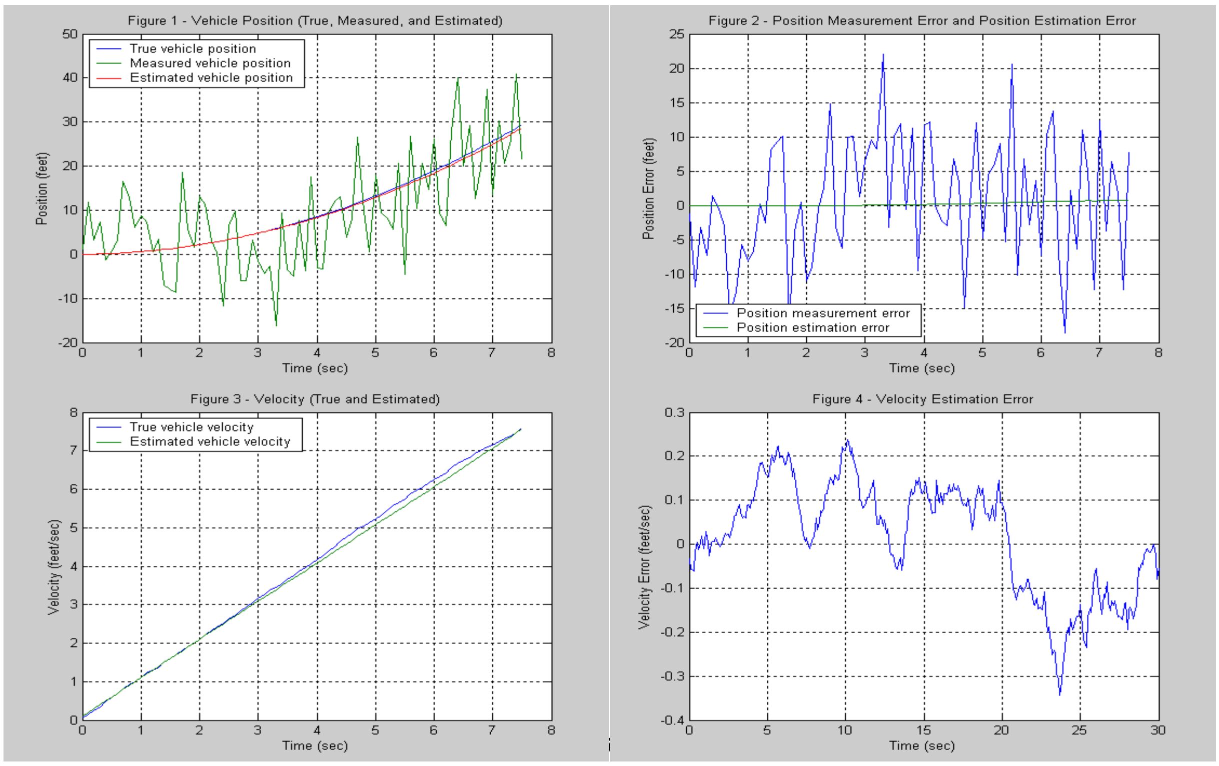

Figure ٦ Execution results for Duration = ٣٠ seconds, step size = ٠,١ seconds, position measurement noise =١٠ ft, and acceleration noise ٠,٢ ft/sec^{٢}.

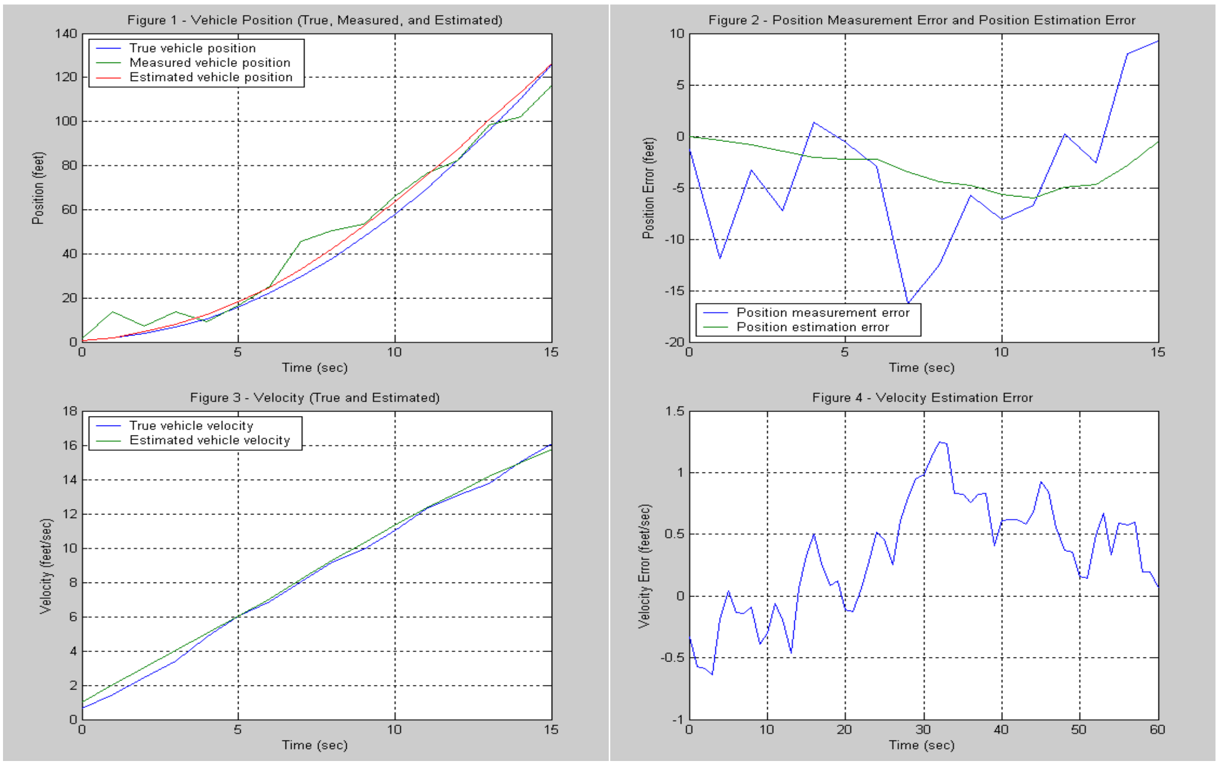

Figure ٧ Execution results for Duration = ٦٠ seconds, step size = ١ seconds, position measurement noise =١٠ ft, and acceleration noise ٠,٢ ft/sec^{٢}.

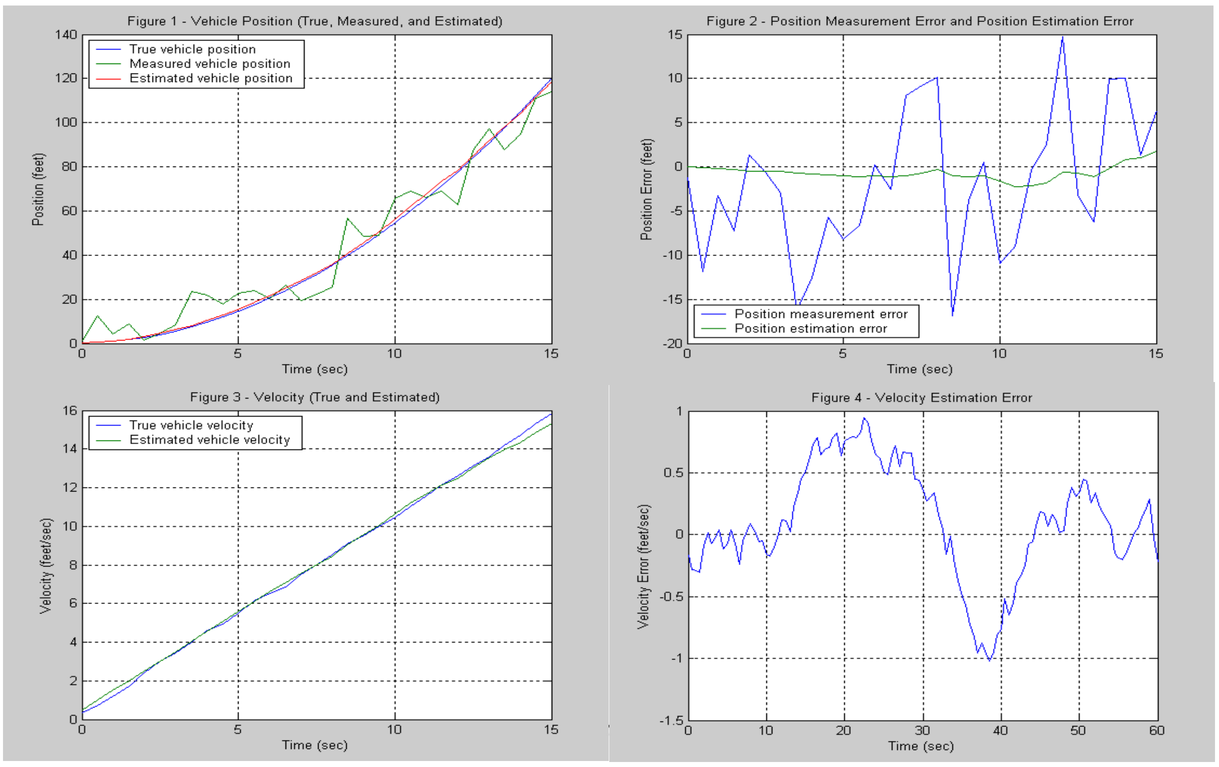

Figure ٨ Execution results for Duration = ٦٠ seconds, step size = ٠,٥ seconds, position measurement noise =١٠ ft, and acceleration noise ٠,٢ ft/sec٢.

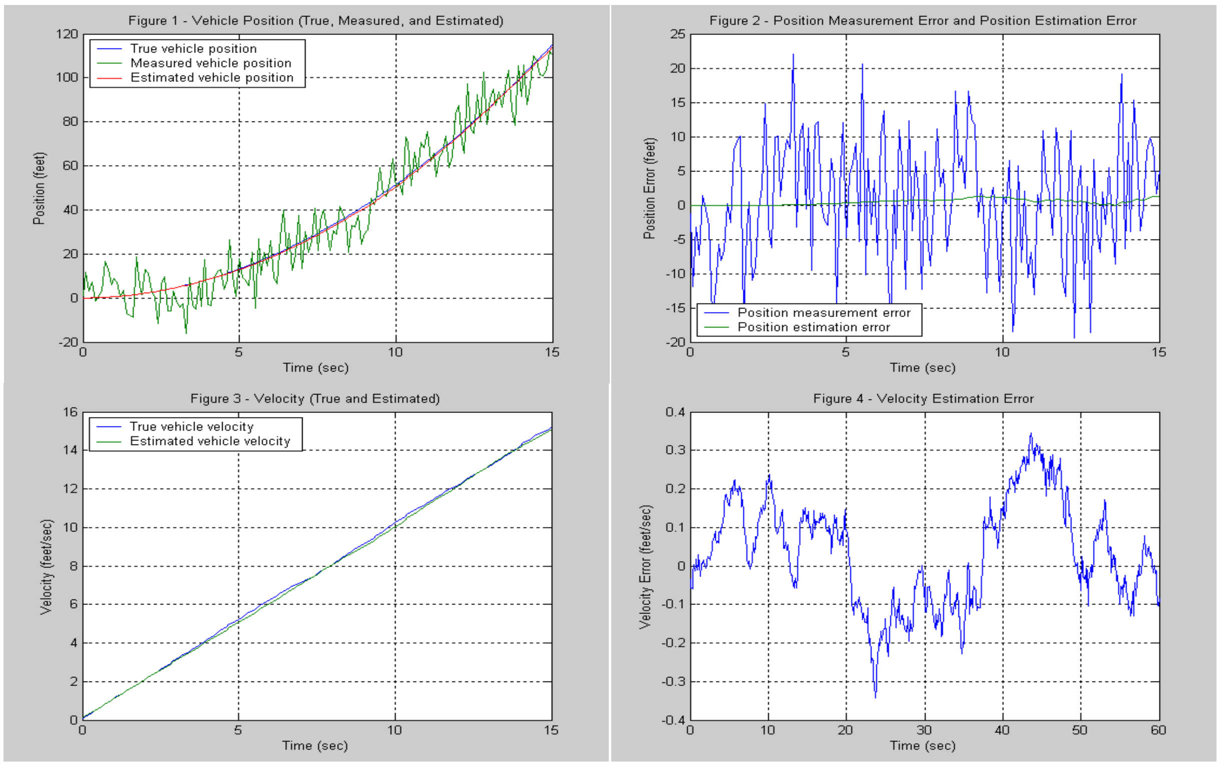

Figure ٩ Execution results for Duration = ٦٠ seconds, step size = ٠,١ seconds, position measurement noise =١٠ ft, and acceleration noise ٠,٢ ft/sec٢.

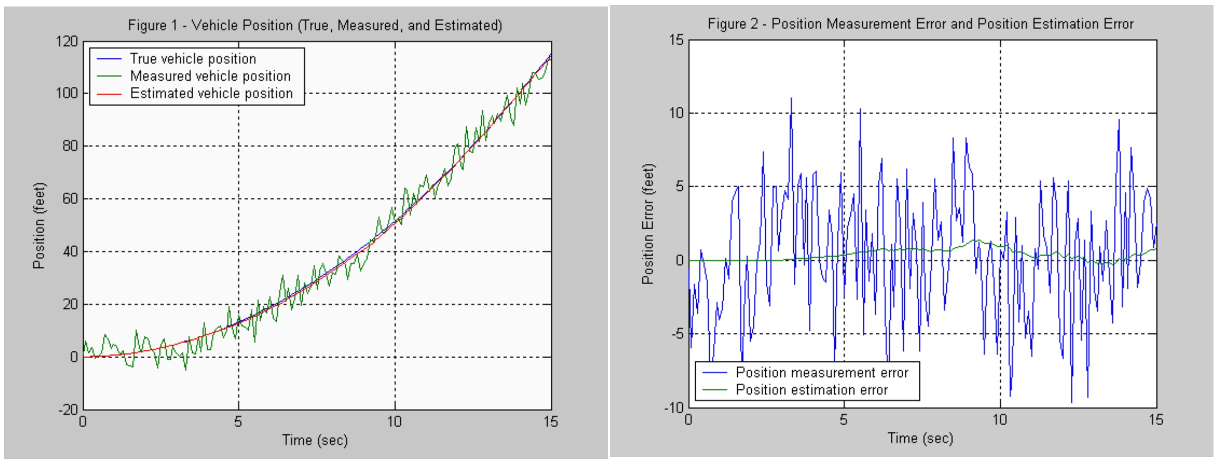

Figure ١٠ Execution results for Duration = ٦٠ seconds, step size = ٠,١ seconds, position measurement noise =٥ ft, and acceleration noise ٠,٢ ft/sec^{٢}.

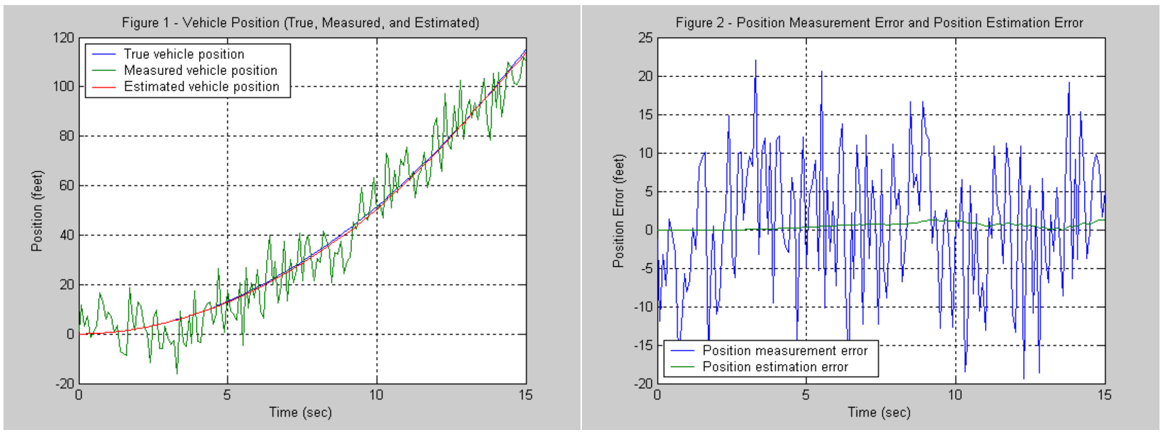

Figure ١١ Execution results for Duration = ٦٠ seconds, step size = ٠,١ seconds, position measurement noise =١٠ ft, and acceleration noise ٠,٢ ft/sec٢.

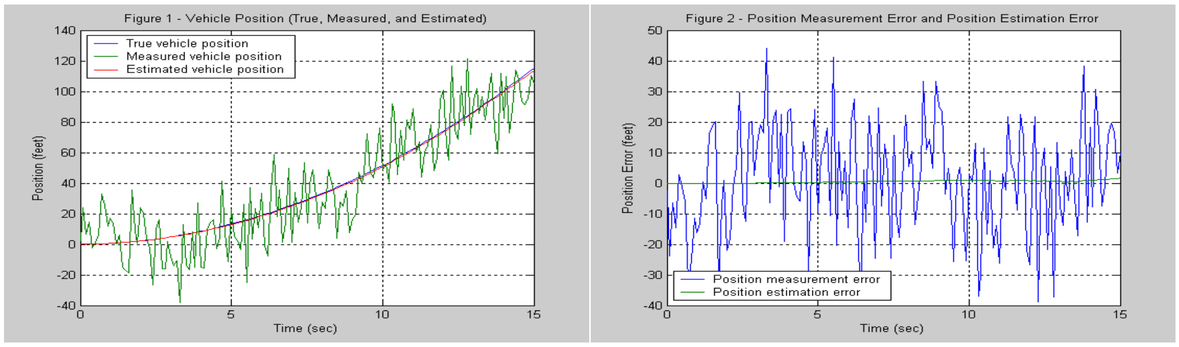

Figure ١٢ Execution results for Duration = ٦٠ seconds, step size = ٠,١ seconds, position measurement noise =٢٠ ft, and acceleration noise ٠,٢ ft/sec^{٢}.

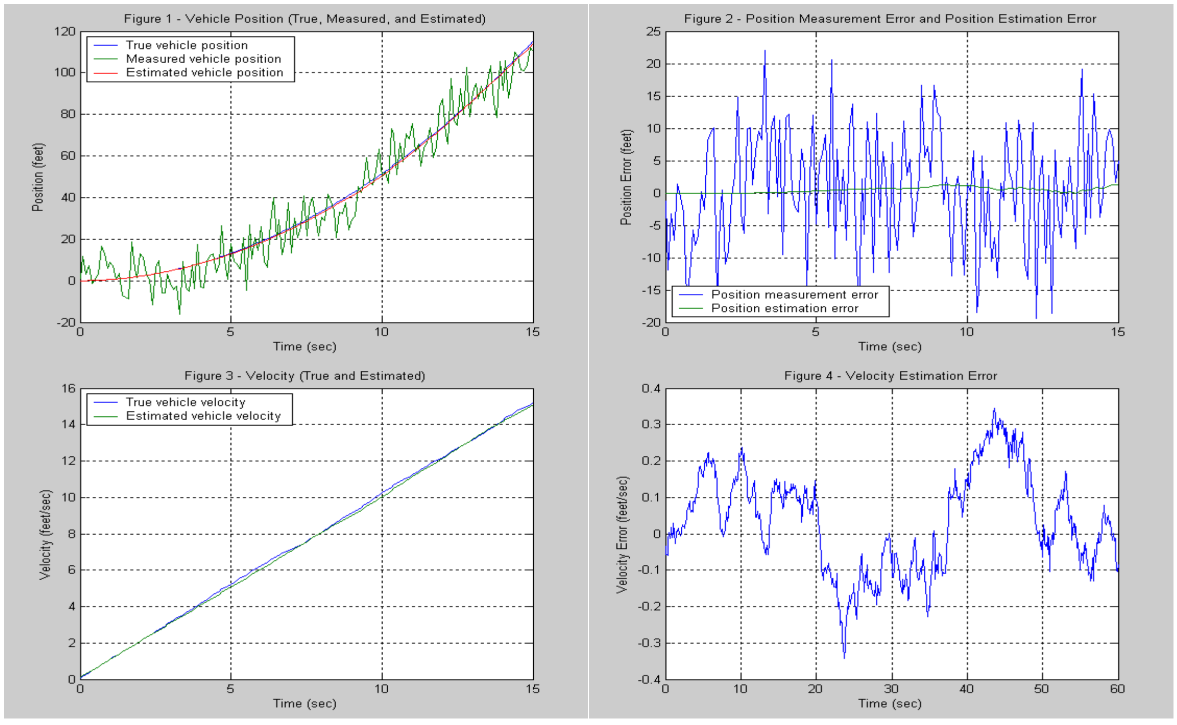

Figure ١٣ Execution results for Duration = ٦٠ seconds, step size = ٠,١ seconds, position measurement noise =١٠ ft, and acceleration noise = ٠,٢ ft/sec^{٢}.

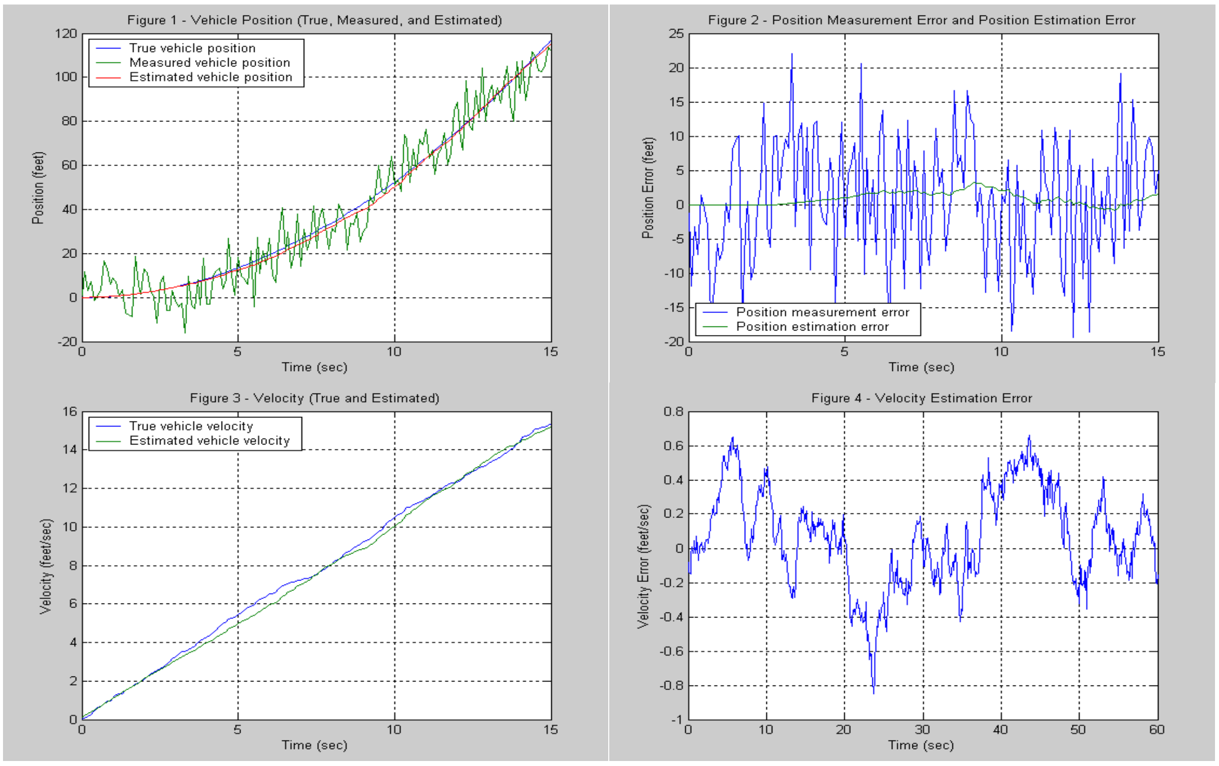

Figure ١٤ Execution results for Duration = ٦٠ seconds, step size = ٠,١ seconds, position measurement noise =١٠ ft, and acceleration noise = ٠,٥ ft/sec^{٢}.

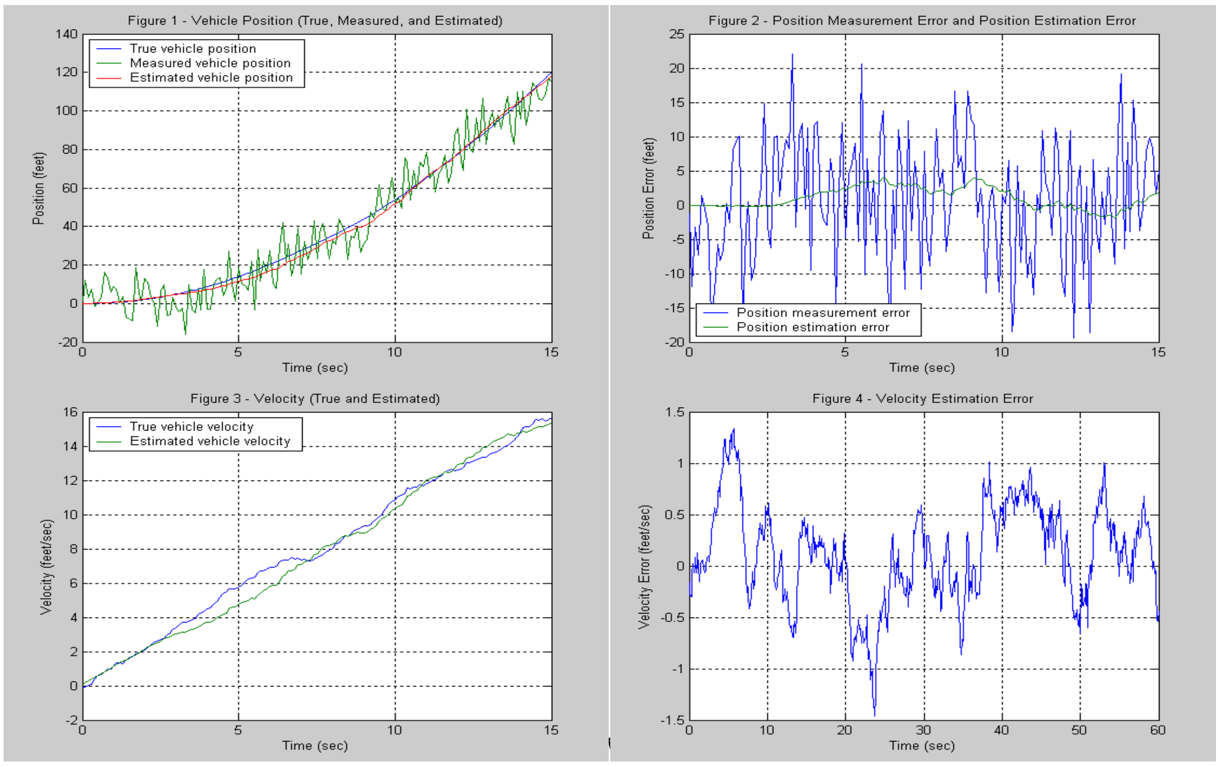

Figure ١٥ Execution results for Duration = ٦٠ seconds, step size = ٠,١ seconds, position measurement noise =١٠ ft, and acceleration noise = ١ ft/sec٢.

٦. Conclusions

There are several points concluded from this work which are:

١- By decreasing the step size for different durations, the performance of discrete Kalman filter will be improved since decreasing the step size will increase the accuracy of estimation process.

٢- By increasing position measurement noise, the measured position will be degraded since the error in measuring process will be high in this case.

٣- By increasing acceleration noise, the performance of discrete Kalman filter will be degraded since acceleration noise will affect the process noise covariance (S_w) and then P_k. This causes the degradation to the whole estimation process.

References

١. Moriya, N., "Primer to Kalman Filtering: A Physicist Perspective", New York: Nova Science Publishers, Inc. ISBN ٩٧٨-١-٦١٦٦٨-٣١١-٥, ٢٠١١.

٢. Brown, R. G., & Hwang, P. Y. C., "Introduction to Random Signals and Applied Kalman Filtering: with MATLAB Exercises and Solutions", Wiley & Sons, Inc, ١٩٩٦.

٣. Simon, D., "Optimal State Estimation: Kalman, H-infinity, and nonlinear approaches", John Wiley & Sons, Inc, ٢٠٠٦.

٤. Jacobs, O. L. R., "Introduction to Control Theory", Second ed., Oxford University Press, ١٩٩٣.

٥. Grewal, M., S., & Andrews, A., P., "Kalman Filtering Theory and Practice Using MATLAB", New York, NY USA: John Wiley & Sons, Inc, ٢٠٠٨.

٦. Haykin, S., "Adaptive Filter Theory", Prentice Hall, ٢٠٠٢.

٧. Zarchan, P., & Musoff, H., "Fundamentals of Kalman Filtering: A Practical Approach", John Wiley & Sons, Inc, ٢٠٠٥.

٨. Chui, Charles K.; Chen, Guanrong, "Kalman Filtering with Real-Time Applications", Springer Series in Information Sciences. ١٧ (٤th ed.), New York: Springer. pp. ٢٢٩. ISBN ٩٧٨-٣-٥٤٠-٨٧٨٤٨-٣, ٢٠٠٩.

٩. Manolakis, D.G., "Statistical and Adaptive signal processing", Artech House,١٩٩٩.

١٠. Z. Ghahramani and G. Hinton, "Parameter estimation for linear dynamical systems", Tech. Rep., ١٩٩٦.