

ENPM667—Control of Robotic Systems

Project 1: Unscented Kalman Filter

Trevian Jenkins

Maryland Applied Graduate Engineering
University of Maryland, College Park
College Park, Maryland, USA
trj0011@umd.edu

Markose Jacob

Maryland Applied Graduate Engineering
University of Maryland, College Park
College Park, Maryland, USA
markj11@umd.edu

27 November 2020

Abstract

The Kalman filter is a widely-used technique for extracting meaningful information out of input data in the presence of disturbances. While the Kalman filter succeeds in situations where the system model is linear, it fails to approximate the true data correctly beyond the range of linearity. The extended Kalman filter provides a generalization of the Kalman filter for use in nonlinear systems, and it performs well when the disturbance is normally distributed, which is usually assumed. However, the extended Kalman filter suffers from bias when the data falls under some other statistical distribution. Such a change in distribution may for example occur, as we will see, from switching from polar to Cartesian coordinates. The unscented Kalman filter circumvents this issue by transforming the distribution along with the data. This report summarizes our implementation of the unscented Kalman filter and describes its advantage over the linear and extended Kalman filters, as well as plotting some examples.

Contents

1	Introduction	3
2	Mathematical Theory	3
2.1	Linear Kalman Filter	3
2.2	Extended Kalman Filter	5
2.3	Complex-Step Derivative Approximation	6
2.4	Unscented Kalman Filter	7
3	Results	8
3.1	Example Sinusoidal Wave	8
3.2	Coordinate System Conversion	8
3.3	Vehicular Trajectory Example	9
4	Conclusion	13
4.1	Future work	14
4.2	Acknowledgements	15

List of Figures

1	Unscented Kalman filter for sinusoidal wave.	9
2	Transformation of coordinates from polar to Cartesian, and the effect on the mean and covariance.	10
3	Trajectory of reentry of a vehicle into the atmosphere of Earth, obtained from equations of motion indicated in 3.3. Simulated from Fig. 8 in Julier and Uhlmann [1].	12
4	Result of estimated trajectory versus true trajectory.	13
5	Results of our simulation, as described in section 3.3.	14

1 Introduction

The Kalman filter was popularized by Rudolf Kalman in 1960, and it has become the *de facto* standard of statistical estimation and prediction when dealing with data that contains disturbances, such as that of robotic sensors. In a linear system, or a subset of data that can be linearized within a certain domain, the basic linear Kalman filter can sufficiently estimate the true data. However, when the system is generally nonlinear, the standard Kalman filter quickly loses accuracy when estimating the values of state variables. Thus, we seek to use a generalization the Kalman filter for use in nonlinear systems.

We referred to "New extension of the Kalman filter to nonlinear systems," by Julier and Uhlmann, to implement the unscented Kalman filter. In this report, we will first describe the mathematical theory behind the linear, extended, and unscented Kalman filters. We will use a simple sinusoidal example to illustrate the unscented Kalman filter in use. We will also replicate Fig. 1 and the space vehicle trajectory example from the paper to show the practical use of the unscented Kalman filter.

2 Mathematical Theory

2.1 Linear Kalman Filter

Let us first look at the standard Kalman filter, or the *linear* Kalman Filter. When using the Kalman filter, we desire to regularly update our estimation of the state variables based on the previous estimation of the state variables, the uncertainty of the accuracy of our current state vector, and the interval of time Δt_P between predictions. Every time the value of the state vector is predicted, the uncertainty, which is stored in a covariance matrix \mathbf{P} , grows, due to the fact that we are effectively extrapolating the current position since any information was last received. We will use the example of a sine wave to illustrate. The equations of motion we will use for this example are as follows:

$$x(t) = \alpha \sin \omega t + \beta \cos \omega t \quad (1)$$

$$\frac{d}{dt}x(t) = \dot{x}(t) = \alpha\omega \cos \omega t - \beta\omega \sin \omega t \quad (2)$$

$$\frac{d^2}{dt^2}x(t) = \ddot{x}(t) = -\alpha\omega^2 \sin \omega t - \beta\omega^2 \cos \omega t \quad (3)$$

We may linearize equations (1) through (3) by using the small angle approximation as follows. If the value of x is assumed to be small (< 0.2 rad), then $\sin x \approx x$. The small angle approximation is derived from the Maclaurin series approximation:

$$f(x) = \sum_{k=0}^{\infty} \frac{x^k f^{(k)}(x)}{k!}$$

The Maclaurin series for the sine function is

$$\begin{aligned}\sin x &= \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k+1}}{(2k+1)!} \\ &= x - \frac{x^3}{6} + \frac{x^5}{120} - \frac{x^7}{5040} + \dots \\ &\approx x,\end{aligned}$$

with the final approximation neglecting higher-order terms. A similar derivation is used to approximate the cosine function as

$$\begin{aligned}\cos x &= \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k}}{(2k)!} \\ &= 1 - \frac{x^2}{2} + \frac{x^4}{24} - \frac{x^6}{720} + \dots \\ &\approx 1 - \frac{x^2}{2}.\end{aligned}$$

For the linear Kalman filter, it is necessary that we make the slightly less accurate, but still reasonable approximation,

$$\cos x \approx 1.$$

This linear approximation is reasonable for many applications and is a precursor for the use of the linear Kalman filter. With the small-angle approximation, the equations of motion then become

$$\begin{aligned}x(t) &\approx \alpha t + \beta \\ \dot{x}(t) &\approx \alpha\omega - \beta\omega t \\ \ddot{x}(t) &\approx -\alpha\omega^2 t - \beta\omega^2 \approx -\omega^2 x\end{aligned}\tag{4}$$

Using our equations of motion and the small angle approximation, we may define the state vector \mathbf{x} as

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}.\tag{5}$$

The state transition matrix A may then be defined as follows:

$$\mathbf{A} = \begin{bmatrix} 1 & \Delta t \\ -\omega^2 \Delta t & 1 \end{bmatrix}\tag{6}$$

The measurement matrix \mathbf{H} describes the relation between a new measurement z and the state vector \mathbf{x} . For example, if we have a system that is capable of measuring x but not \dot{x} , then h would be $\begin{bmatrix} 1 & 0 \end{bmatrix}$.

With equations (5) and (6), we find that the next (*a priori*) values of the state vector and covariance matrix are predicted to be

$$\begin{aligned}\mathbf{x}_n^- &= \mathbf{A}\mathbf{x}_{n-1} \\ \mathbf{P}_n^- &= \mathbf{A}\mathbf{P}_{n-1}\mathbf{A}^T + \mathbf{Q},\end{aligned}\tag{7}$$

where \mathbf{Q} is the covariance matrix describing the process noise (from random disturbances, etc.). When new data is received, the measurement step updates \mathbf{x} and \mathbf{P} as follows:

$$\begin{aligned}\mathbf{K}_n &= \mathbf{P}_n^- \mathbf{H}^T (\mathbf{H}\mathbf{P}_n^- \mathbf{H}^T + \mathbf{R})^{-1} \\ \mathbf{x}_n &= \mathbf{x}_n^- + \mathbf{K}_n(z_n - \mathbf{H}\mathbf{x}_n^-) \\ \mathbf{P}_n &= (\mathbf{I} - \mathbf{K}_n\mathbf{H})\mathbf{P}_n^-\end{aligned}\tag{8}$$

2.2 Extended Kalman Filter

In the case of the extended Kalman filter, we run the prediction and measurement steps as before. The major change, however, is that instead of using matrices to define the state transitions, we simply use a vector function. Equations (1) through (3) dictate that the next (*a priori*) value of the state vector is predicted to be

$$\begin{aligned}\mathbf{x}_n^- &= f(\mathbf{x}_{n-1}) \\ &= \begin{bmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{bmatrix} \\ &= \begin{bmatrix} x_1 + x_2\Delta t \\ x_2 - \omega^2 x_1\Delta t \end{bmatrix},\end{aligned}\tag{9}$$

and when this prediction is made, the uncertainty matrix becomes

$$\mathbf{P}_n^- = \mathbf{A}_n\mathbf{P}_{n-1}\mathbf{A}_n^T + \mathbf{W}_n\mathbf{Q}_{n-1}\mathbf{W}_n^T,\tag{10}$$

where \mathbf{A} is the Jacobian matrix

$$\mathbf{A}_{ij} = \frac{\partial f_i}{\partial x_j}(\mathbf{x}_{n-1}),$$

\mathbf{Q} is the covariance matrix describing the process noise w , and \mathbf{W} is the Jacobian matrix

$$\mathbf{W}_{ij} = \frac{\partial f_i}{\partial w_j}(\mathbf{x}_{n-1}).$$

The function $h(\mathbf{x})$ describes the relation between a new measurement z and the state vector \mathbf{x} . Whenever new measurements are retrieved, such as azimuth information in our case, the next (*a posteriori*) value of the state matrix is updated to be

$$\mathbf{x}_n = \mathbf{x}_n^- + \mathbf{K}_n[z_n - h(\mathbf{x}_n^-)],\tag{11}$$

where

$$\mathbf{K}_n = \mathbf{P}_n^- \mathbf{H}_n^T (\mathbf{H}_n \mathbf{P}_n \mathbf{H}_n^T + \mathbf{V}_n \mathbf{R}_n \mathbf{V}_n^T)^{-1},$$

\mathbf{H} is the Jacobian matrix

$$\mathbf{H}_{ij} = \frac{\partial h_i}{\partial x_j}(\mathbf{x}_n),$$

\mathbf{R} is the covariance matrix describing the measurement noise v , and \mathbf{V} is the Jacobian matrix

$$\mathbf{V}_{ij} = \frac{\partial h_i}{\partial v_j}(\mathbf{x}_n).$$

With the arrival of new measurements, uncertainty always decreases, and the new uncertainty matrix \mathbf{P} becomes

$$\mathbf{P}_n = (\mathbf{I} - \mathbf{K}_n \mathbf{H}_n) \mathbf{P}_n^-. \quad (12)$$

The $(-)$ superscript in the \mathbf{x}^- vector and \mathbf{P}^- matrices indicate that those values are *a priori* values that were calculated from the predict step, before the arrival of measurement data. We repeat the cycle of obtaining measurements of the current state vector and using those measurements to infer about the next values of the state vector. The output of the Kalman filter provides an improved estimate over the raw sensor data, which is to be used as e.g. the feedback within a control system, or for visualization.

2.3 Complex-Step Derivative Approximation

We utilized the method described in this section to compute the Jacobian matrices for the Kalman filter. Traditionally, the derivative of a function $f(x)$ is estimated using the finite difference approximation

$$f'(x) \approx \frac{f(x+h) - f(x)}{h} \quad (13)$$

for, say, a small stepsize $h \approx 10^{-7}$. However, we wish to avoid cancellation error arising from taking differences of such small values. Thus, the partial derivatives of the aforementioned Jacobian functions are computed using the complex-step derivative approximation described as follows. The derivative of $f(x)$ with respect to x is calculated as

$$\frac{\partial f}{\partial x} = \frac{\Im[f(x+ih)]}{h}, \quad (14)$$

where $\Im[z]$ returns the imaginary part of the complex number z . The complex step method results from the Taylor series approximation about a real point x :

$$f(x+ih) = f(x) + ihf'(x) - \frac{h^2 f''(x)}{2!} - \frac{ih^3 f'''(x)}{3!} + \dots \quad (15)$$

Taking the imaginary part of equation (15) and dividing by h results in

$$\begin{aligned}
f'(x) &= \frac{\Im[f(x+ih)]}{h} + \frac{h^2 f'''(x)}{3!} + \frac{h^4 f^{(5)}(x)}{5!} + \dots \\
&= \frac{\Im[f(x+ih)]}{h} + \mathcal{O}(h^2) \\
&\approx \frac{\Im[f(x+ih)]}{h}.
\end{aligned} \tag{16}$$

The final result truncates the higher-order terms, giving a truncation error $\mathcal{O}(h^2)$. Since this calculation does not involve a subtraction operation, it is not prone to cancellation errors resulting from subtractions. Therefore, we consider the complex step method superior to the more traditional derivative approximation in equation (13) for small stepsize h , hence our use in the extended Kalman filter.

2.4 Unscented Kalman Filter

The basic setup of the unscented Kalman filter is similar to the extended kalman filter. We still run the prediction and measurement steps as we always have. However, we now implement what is known as an *unscented transformation*. This operation transforms the mean and covariance information about a distribution along with the data itself.

The following steps are performed to make this transformation happen:

1. Creating the sigma points through a sigma point selection algorithm.
2. Calculating the transformed set:

$$\hat{x}_{a,n} = f[x_{a,n}]$$

3. Predicting the mean:

$$\hat{\mu}_{a,n} = \sum_{i=0}^p W^{(i)} \hat{x}_{a,n}^{(i)}$$

4. Predicting the covariance:

$$\hat{K}_{a,n} = \sum_{i=0}^p W^{(i)} (\hat{x}_{a,n}^{(i)} - \hat{\mu}_{a,n})(\hat{x}_{a,n}^{(i)} - \hat{\mu}_{a,n})^T$$

5. Instantiating prediction points:

$$y_n^{(i)} = g[x_{a,n}^{(i)}]$$

6. Predicting the observation:

$$y_n = g[y_n^{(i)}]$$

7. Calculating the innovation covariance:

$$\hat{S}_n = \sum_{i=0}^p W^{(i)} (\hat{y}_n^{(i)} - \hat{y}_n) (\hat{y}_n^{(i)} - \hat{y}_n)^T$$

8. Calculating the cross covariance matrix:

$$\hat{K}_n^{xy} = \sum_{i=0}^p W^{(i)} (\hat{x}_n^{(i)} - \hat{\mu}_n) (\hat{x}_n^{(i)} - \hat{\mu}_n)^T$$

9. The update is performed as per the typical Kalman filter operation:

$$\begin{aligned}\mu_n &= \hat{\mu}_n + W_n \nu_n \\ K_n &= \hat{K}_n - W_n \hat{S}_n W_n^T \\ \nu_n &= y_n - \hat{y}_n \\ W_n &= \hat{K}_n^{xy} \hat{S}_n^{-1}\end{aligned}$$

From step (9) above, we obtain our new *a posteriori* estimate of the mean value of the state vector μ_n and error covariance matrix K_n .

3 Results

3.1 Example Sinusoidal Wave

The results we achieved are as follows. From our sinusoidal example described before, we illustrate the true curve and the filtered data in Fig. 1. The true value of the wave, the measurements, and the filtered value of the wave are indicated in the plot, with the filtered value of the wave at any given time instant t in reasonable correspondence with the true value at that time.

3.2 Coordinate System Conversion

To illustrate the effects on nonlinear transformation, we will show the transformation of data points representing location data from a sensor tracking a target placed at (x, y) coordinates $(0, 1)$. The sensor used in the example from the original paper returned a mean radius of 1.0 m, with a standard deviation of 0.02 m. The mean angle returned by the sensor was $\frac{\pi}{2}$ rad, while the standard deviation of the angle was $\frac{\pi}{20}$ rad. This data was converted into Cartesian coordinates as

$$\begin{aligned}x &= r \cos \theta \\ y &= r \sin \theta\end{aligned}\tag{17}$$

The data points are plotted in the left plot, and the irregular distribution resulting from the coordinate transformation is apparent.

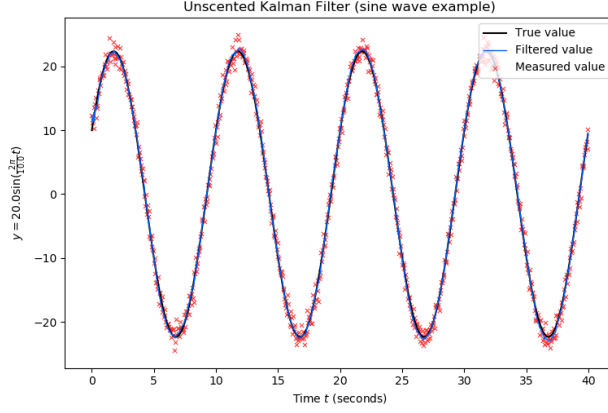


Figure 1: Unscented Kalman filter for sinusoidal wave.

On the right are plots containing the statistical information contained within the data. The x symbol indicates the true mean of the data, and the dashed ellipse is the true covariance ellipse. The linearized version of the sensor data has a mean indicated by the o symbol, and the dotted ellipse is the corresponding covariance ellipse. As one can see, the data after a linearization suffers from bias created by the linearization of the data points without transforming the statistical data as well.

3.3 Vehicular Trajectory Example

For the practical example we decided to illustrate, we attempted to replicate the results of Section VI—“Applying the UT to Recursive Estimation”—from the original paper by Julier and Uhlmann [1]. This section details the estimation of the state of a vehicle as it is reentering Earth’s atmosphere. The vehicle is being observed by a radar on Earth’s surface, near the projected landing location.

The following state variables are tracked by the filter:

- x_1 : The x -component of the position of the vehicle
- x_2 : The y -component of the position of the vehicle
- x_3 : The x -component of the velocity of the vehicle
- x_4 : The y -component of the velocity of the vehicle
- x_5 : The aerodynamic coefficient, which is a property of the aerodynamics of the vehicle

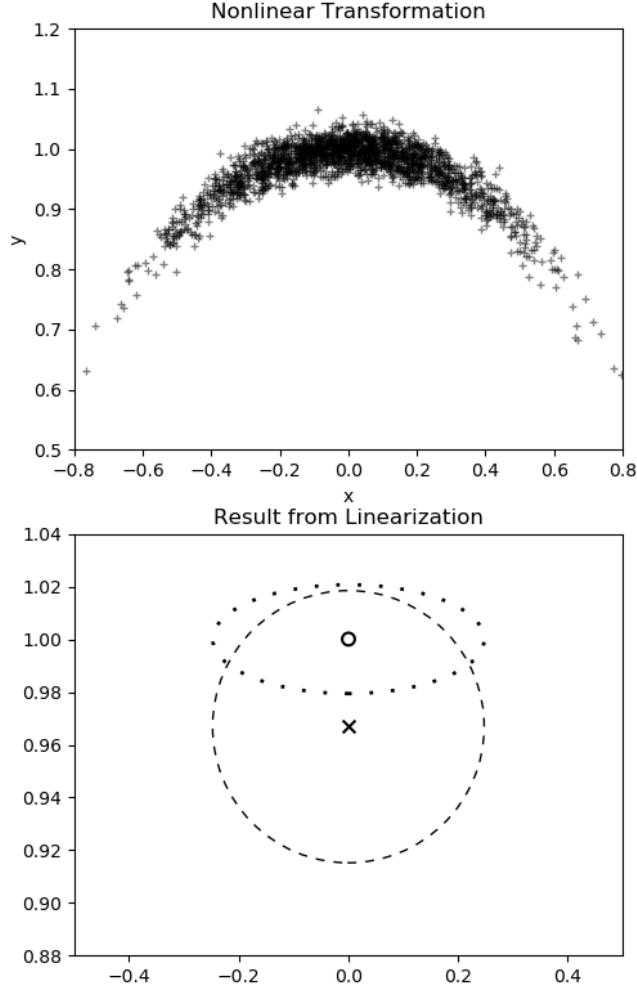


Figure 2: Transformation of coordinates from polar to Cartesian, and the effect on the mean and covariance.

The vehicle has the following state dynamics:

$$\begin{aligned}
 \dot{x}_1(k) &= x_3(k) \\
 \dot{x}_2(k) &= x_4(k) \\
 \dot{x}_3(k) &= D(k)x_3(k) + G(k)x_1(k) + v_1(k) \\
 \dot{x}_4(k) &= D(k)x_4(k) + G(k)x_2(k) + v_2(k) \\
 \dot{x}_5(k) &= v_3(k)
 \end{aligned} \tag{18}$$

The force $D(k)$ is the drag force at iteration k . The force $G(k)$ is the force due to gravity, and v_k is the process noise. The distance to the vehicle from the

center of Earth is the function $R(k) = \sqrt{x_1^2(k) + x_2^2(k)}$, while the speed of the vehicle is $V(k) = \sqrt{x_3^2(k) + x_4^2(k)}$. The drag and gravitational forces, as well as other important variables and functions, are as follows:

$$\begin{aligned}
D(k) &= -\beta(k) \exp\left(\frac{R_0 - R(k)}{H_0}\right) V(k) \\
G(k) &= -\frac{Gm_0}{r^3(k)} \\
\beta(k) &= \beta_0 \exp(x_5(k)) \\
\mathbf{v}(k) &= [v_1(k) \quad v_2(k) \quad v_3(k)]^T \\
&= [2.4064 \times 10^{-5} \quad 2.4064 \times 10^{-5} \quad 0]^T
\end{aligned} \tag{19}$$

Some important constants are defined below:

$$\begin{aligned}
\beta_0 &= 0.59783 \\
H_0 &= 13.406 \\
Gm_0 &= 3.9860 \times 10^5 \text{ (G constant} \times \text{mass of Earth)} \\
R_0 &= 6374 \text{ (Radius of Earth)}
\end{aligned} \tag{20}$$

One important note: The value of β_0 was preceded by a “−” sign in the paper, which is incorrect since it results in a double negative, given that the drag force $D(\mathbf{k})$ already contains the appropriate negative sign. This issue gave us some trouble in simulating an accurate model of the system dynamics until we figured out the error.

Using the dynamics of equation (18), we modeled the trajectory of the vehicle in Fig. 3, which matches closely with the original paper as expected. We repeated iterations of the dynamics until the vehicle landed on the surface.

Like in the original paper, we assigned to the system the initial conditions

$$\mathbf{x}(0) = [6500.4 \quad 349.14 \quad -1.8093 \quad -6.7967 \quad 0.6932]^T$$

and

$$P(0) = \begin{bmatrix} 10^{-6} & 0 & 0 & 0 & 0 \\ 0 & 10^{-6} & 0 & 0 & 0 \\ 0 & 0 & 10^{-6} & 0 & 0 \\ 0 & 0 & 0 & 10^{-6} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

while we let the unscented Kalman filter assume the initial conditions

$$\mu_0 = [6500.4 \quad 349.14 \quad -1.8093 \quad -6.7967 \quad 0]^T$$

and

$$K_0 = \begin{bmatrix} 10^{-6} & 0 & 0 & 0 & 0 \\ 0 & 10^{-6} & 0 & 0 & 0 \\ 0 & 0 & 10^{-6} & 0 & 0 \\ 0 & 0 & 0 & 10^{-6} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

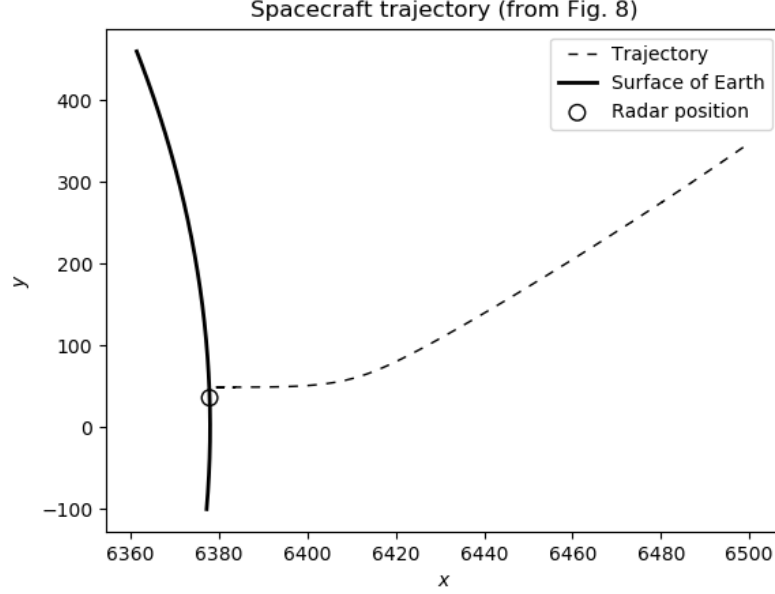


Figure 3: Trajectory of reentry of a vehicle into the atmosphere of Earth, obtained from equations of motion indicated in 3.3. Simulated from Fig. 8 in Julier and Uhlmann [1].

indicating that there is relative initial certainty of the value of every variable except the aerodynamic coefficient. The assumed process noise of the filter was adapted from the actual process noise $v(k)$,

$$Q(k) = \begin{bmatrix} 2.4064 \times 10^{-5} & 0 & 0 \\ 0 & 2.4064 \times 10^{-5} & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

The measurements and updates were performed at a frequency of 10 Hz, and the simulation was performed for a duration of 200 seconds, for a total of 1000 iterations.

For visual reference, we provided a plot of the estimated trajectory of the vehicle, compared to the vehicle's true trajectory, in Fig. 4. Our results shown in Fig. 5 are of the mean squared error and variance for the unscented Kalman filter for x_1 , x_3 , and x_5 , which represent the vehicle's x -position, x -velocity, and aerodynamic coefficient, respectively. We only plotted the error of the mean, as well as the covariance, from the unscented Kalman filter. As seen in Fig. 4, our Kalman filter implementation tracks the trajectory of the vehicle reasonably well up to the termination of the estimation program, represented by the red \times in the plot. We could not quite replicate the error estimation of the original simulation, likely due to differences in the way we calculated and/or

reported our error calculations. This divergence could also be a result of the difference in randomization between the original simulation and our implementation. Nonetheless, our trajectory tracking via unscented Kalman filter has performed similarly well.

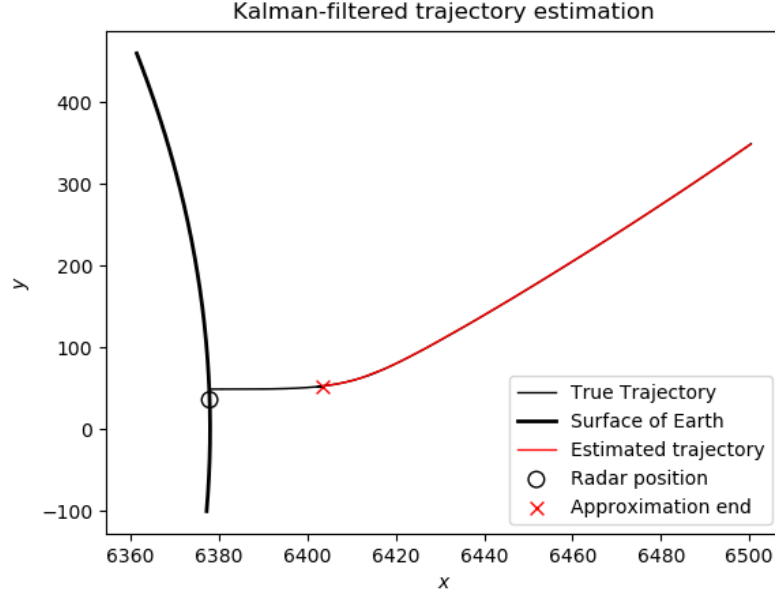
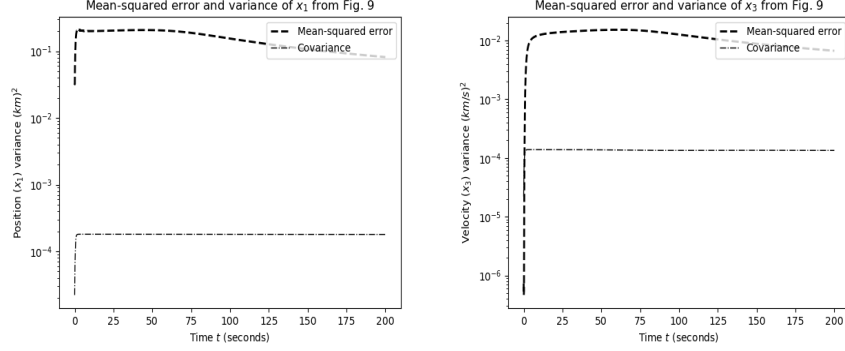


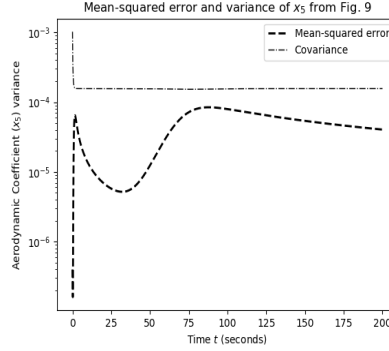
Figure 4: Result of estimated trajectory versus true trajectory.

4 Conclusion

This project revolved around the implementation of the unscented Kalman filter for estimation of variables in a nonlinear system with non-Gaussian-distributed noise. Leading to this implementation, we also implemented the standard linear Kalman filter, as well as the extended Kalman filter for nonlinear systems. We also showed a comparison of the linear and unscented Kalman filters with a sinusoidal example in section 3.1. From the original paper by Julier and Uhlmann, we managed to replicate the results of the transformation of statistical data—mean and covariance— shown in this report in Fig. 2 and described in section 3.2. We also managed to replicate results of the vehicle trajectory estimation in section 3.3.



(a) Error of mean and covariance of x_1 . (b) Error of mean and covariance of x_3 .



(c) Error of mean and covariance of x_5 .

Figure 5: Results of our simulation, as described in section 3.3.

4.1 Future work

This Kalman filter implementation has room for improvement, as well as potential for further exploration. The primary aspect that can be improved is the separation of the prediction and measurement stages. As of right now, the prediction and measurement steps are alternated continuously. This process can be generalized for the case of multiple predictions between measurements, as well as, conversely, multiple measurements at once. This update would be practically useful for, say, a sensor that provides few measurements far in between, so that the filter would have to continuously estimate the change in the state vector and error covariance matrix before receiving another measurement. Likewise, in practice, a system may have multiple sensors, with inputs possibly arriving simultaneously, in which case one would wish to run the measurement step multiple times in succession before the need to predict future states.

The original paper by Julier and Uhlmann contained some examples of the use of the unscented transform. Replicating more of the results from their work,

particularly these unscented transformation examples, would provide more understanding of the uses and limits of the unscented transformation.

Finally, readers who seek to learn more about the situations where various estimation strategies succeed over others would benefit from an extensive direct comparison of the Kalman filters. Theoretical and practical examples utilizing different Kalman filters could be the subject of a future project or report.

4.2 Acknowledgements

We would like to thank Dr. Waseem Malik at The University of Maryland, as well as our teaching assistant Srikumar Muralidharan, for giving us a theoretical foundation for completing this project.

References

- [1] S. J. Julier, and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," *Proc. SPIE*, vol. 3068, pp. 182-193, July 1997, doi: 10.1117/12.280797.
- [2] J. R. R. A. Martins, P. Sturdza, and J. J. Alonso, "The complex-step derivative approximation," *ACM Trans. Math. Softw.*, vol. 29, no. 3, pp. 245-262, Sep. 2003, doi: 10.1145/838250.838251
- [3] G. Welch, and G. Bishop, "An introduction to the Kalman filter," Dept. Comp. Sci., Univ. of North Carolina at Chapel Hill, Chapel Hill, NC, USA, Rep. TR 95-041, 2006.