Chris Chamberlain and Mitchell Smith

ASEN 5044
Statistical Estimation for Dynamical Systems
Orbit Determination
December 21, 2017

I. HW 8, Question 2

A.

Find the required CT Jacobians needed to obtain CT linearized model parameters.
The steps to obtain each partial is provided at the end of this document.

$$A_{nom} = \left[\frac{\partial f}{\partial x}\bigg|_{nom}\right] = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}$$

$$x = \begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \quad \rightarrow \quad \dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix}$$

$$f_1 = \dot{x} = x_2$$

$$f_2 = \frac{-\mu x_1}{(x_1^2 + x_3^2)^{3/2}} + u_1 + \tilde{w}_1$$

$$f_3 = \dot{y} = x_4$$

$$f_4 = \frac{-\mu x_3}{(x_1^2 + x_3^2)^{3/2}} + u_2 + \tilde{w}_2$$

$$\frac{\partial f_1}{\partial x_1} = 0 \quad , \quad \frac{\partial f_1}{\partial x_2} = 1$$

$$\frac{\partial f_1}{\partial x_3} = 0 \quad , \quad \frac{\partial f_1}{\partial x_4} = 0$$

$$\frac{\partial f_2}{\partial x_1} = \mu\frac{2x_1^2 - x_3^2}{r^5}$$

$$\frac{\partial f_2}{\partial x_3} = \mu\frac{3x_1 x_3}{r^5}$$

$$\frac{\partial f_2}{\partial x_2} = 0 \quad , \quad \frac{\partial f_2}{\partial x_4} = 0$$

$$\frac{\partial f_3}{\partial x_1} = 0 \quad , \quad \frac{\partial f_3}{\partial x_2} = 0$$

$$\frac{\partial f_3}{\partial x_3} = 0 \quad , \quad \frac{\partial f_3}{\partial x_4} = 1$$

$$\frac{\partial f_4}{\partial x_1} = \mu \frac{3x_1 x_3}{r^5}$$
$$\frac{\partial f_4}{\partial x_3} = \mu \frac{2x_3^2 - x_1^2}{r^5}$$
$$\frac{\partial f_4}{\partial x_2} = 0 \quad , \quad \frac{\partial f_4}{\partial x_4} = 0$$

$$\boxed{A_{nom} = \left[\frac{\partial f}{\partial x}\bigg|_{nom}\right] = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \mu \frac{2x_1^2 - x_3^2}{r^5} & 0 & \mu \frac{3x_1 x_3}{r^5} & 0 \\ 0 & 0 & 0 & 1 \\ \mu \frac{3x_1 x_3}{r^5} & 0 & \mu \frac{2x_3^2 - x_1^2}{r^5} & 0 \end{bmatrix}_{nom}}$$

where $r = \sqrt{x_1^2 + x_3^2}$ and $A \in \mathbb{R}^{4x4}$

$$B_{nom} = \left[\frac{\partial f}{\partial x}\bigg|_{nom}\right] = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \cdots & \frac{\partial f_n}{\partial u_n} \end{bmatrix}$$
$$\frac{\partial f_1}{\partial u_1} = 0 \quad , \quad \frac{\partial f_1}{\partial u_2} = 0$$
$$\frac{\partial f_2}{\partial u_1} = 1 \quad , \quad \frac{\partial f_2}{\partial u_2} = 0$$
$$\frac{\partial f_3}{\partial u_1} = 0 \quad , \quad \frac{\partial f_3}{\partial u_2} = 0$$
$$\frac{\partial f_4}{\partial u_1} = 0 \quad , \quad \frac{\partial f_4}{\partial u_2} = 1$$

$$\boxed{B_{nom} = \left[\frac{\partial f}{\partial x}\bigg|_{nom}\right] = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}_{nom}}$$

where $B \in \mathbb{R}^{4x2}$

$$h_1 = \rho^i(t) = \sqrt{(x_1 - x_s)^2 + (x_3 - y_s)^2}$$
$$h_2 = \dot{\rho}^i(t) = \frac{(x_1 - x_s)(x_2 - \dot{x}_s) + (x_3 - y_s)(x_4 - \dot{y}_s)}{h_1}$$
$$h_3 = \phi^i(t) = \tan^{-1}\left(\frac{x_3 - y_s}{x_1 - x_s}\right)$$
$$C^i_{nom} = \left[\frac{\partial h}{\partial x}\bigg|_{nom}\right] = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_n}{\partial x_1} & \cdots & \frac{\partial h_n}{\partial x_n} \end{bmatrix}$$

where

$$\frac{\partial h_1}{\partial x_1} = \frac{x_1 - x_s}{\sqrt{(x_1 - x_s)^2 + (x_3 - y_s)^2}}$$
$$\frac{\partial h_1}{\partial x_2} = 0$$
$$\frac{\partial h_1}{\partial x_3} = \frac{x_3 - y_s}{\sqrt{(x_1 - x_s)^2 + (x_3 - y_s)^2}}$$
$$\frac{\partial h_1}{\partial x_4} = 0$$

$$\frac{\partial h_2}{\partial x_1} = \frac{x_2 - \dot{x}_s}{\sqrt{(x_1 - x_s)^2 + (x_3 - y_s)^2}} - \frac{(x_1 - x_s)[(x_1 - x_s)(x_2 - \dot{x}_s) + (x_3 - y_s)(x_4 - \dot{y}_s)]}{(\sqrt{(x_1 - x_s)^2 + (x_3 - y_s)^2})^3}$$
$$\frac{\partial h_2}{\partial x_2} = \frac{x_1 - x_s}{\sqrt{(x_1 - x_s)^2 + (x_3 - y_s)^2}}$$
$$\frac{\partial h_2}{\partial x_3} = \frac{x_4 - \dot{y}_s}{\sqrt{(x_1 - x_s)^2 + (x_3 - y_s)^2}} - \frac{(x_3 - y_s)[(x_1 - x_s)(x_2 - \dot{x}_s) + (x_3 - y_s)(x_4 - \dot{y}_s)]}{(\sqrt{(x_1 - x_s)^2 + (x_3 - y_s)^2})^3}$$
$$\frac{\partial h_2}{\partial x_4} = \frac{x_3 - y_s}{\sqrt{(x_1 - x_s)^2 + (x_3 - y_s)^2}}$$
$$\frac{\partial h_3}{\partial x_1} = \frac{y_s - x_3}{(x_s - x_1)^2 + (y_s - x_3)^2}$$
$$\frac{\partial h_3}{\partial x_2} = 0$$
$$\frac{\partial h_3}{\partial x_3} = \frac{x_1 - x_s}{(x_s - x_1)^2 + (y_s - x_3)^2}$$
$$\frac{\partial h_3}{\partial x_4} = 0$$
$$C^i_{nom} \in \mathbb{R}^{3x4}$$

The C_{nom} matrix is not presented here in matrix form, as the equations that go in each cell are too large for it to fit properly in this document. However, if the above definition for C_{nom} is followed, all entries can be found here.

B. Discretize the CT System

$$\tilde{F} = I + \Delta t A_{nom}$$
$$\tilde{G} = \Delta t B_{nom}$$
$$\tilde{H} = C_{nom}$$
$$\tilde{\Omega} = \Delta t \Gamma$$

The CT system can be discretized by using the above equations. However, since A_{nom} is a time varying matrix, and the system does not have a static equilibrium, observability, controllability, and stability can not be discussed.

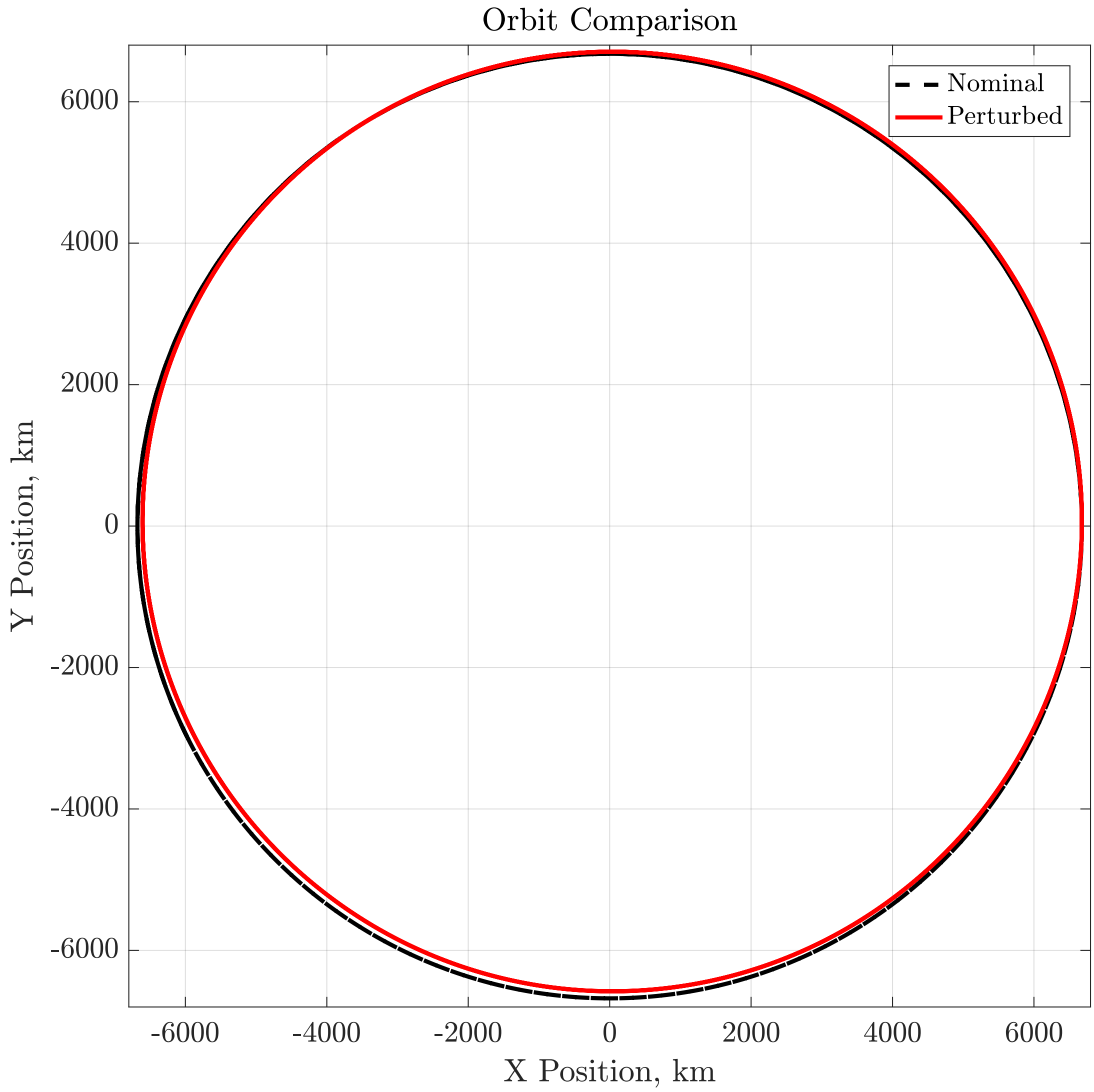
The initial conditions for the nominal orbit are as follows:

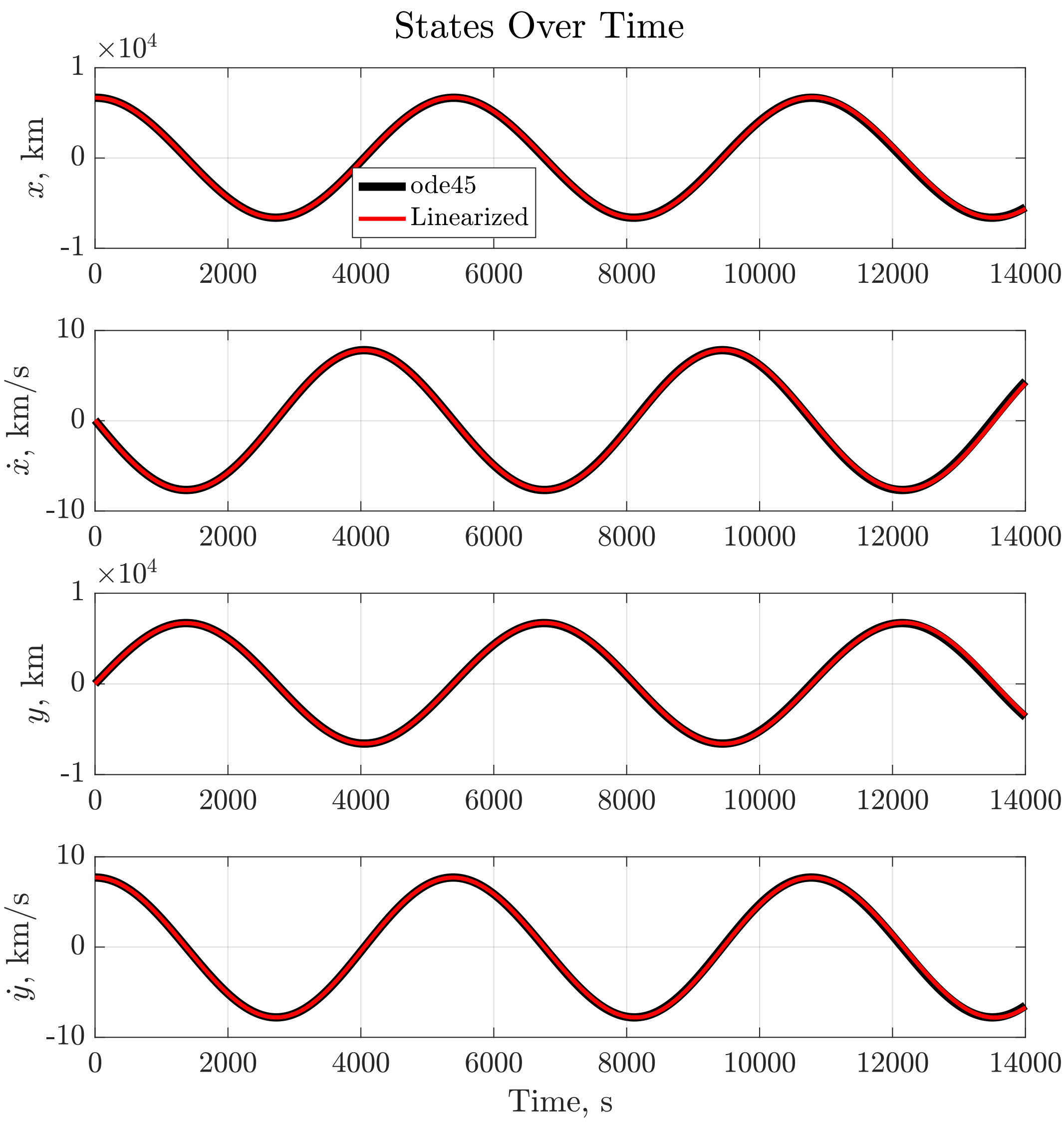
$$\Delta t = 10 \text{ seconds}$$
$$X(0) = 6678 \text{ km} \quad , \quad \dot{X}(0) = 0 \text{km/s}$$
$$Y(0) = 0 \text{ km} \quad , \quad \dot{Y}(0) = r_0 \sqrt{\frac{\mu}{r_0^3}} = 7.7258 \text{ km/s}$$

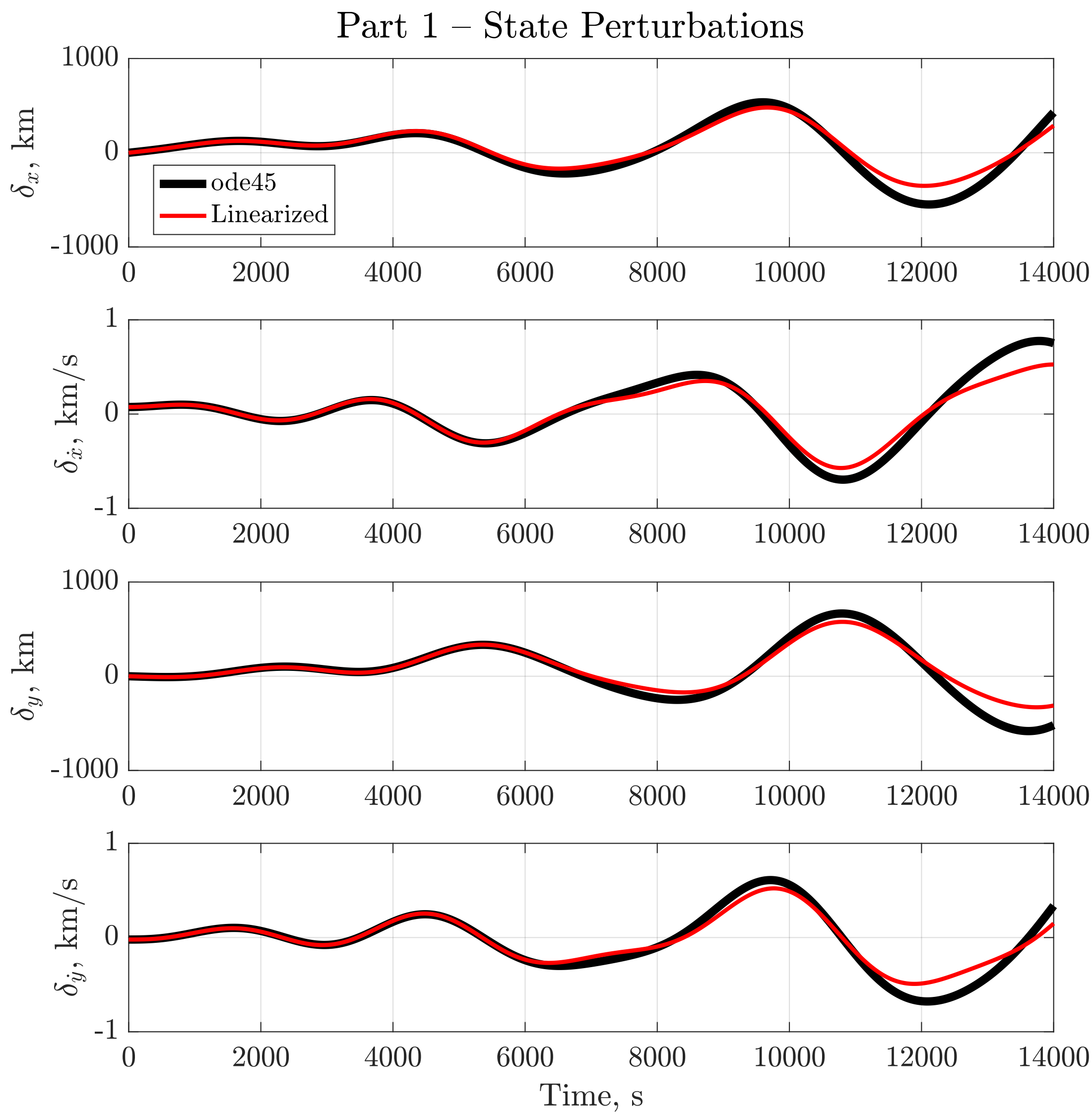
These conditions were used with MATLAB's *ode45* function to create the circular, non-perturbed orbit.

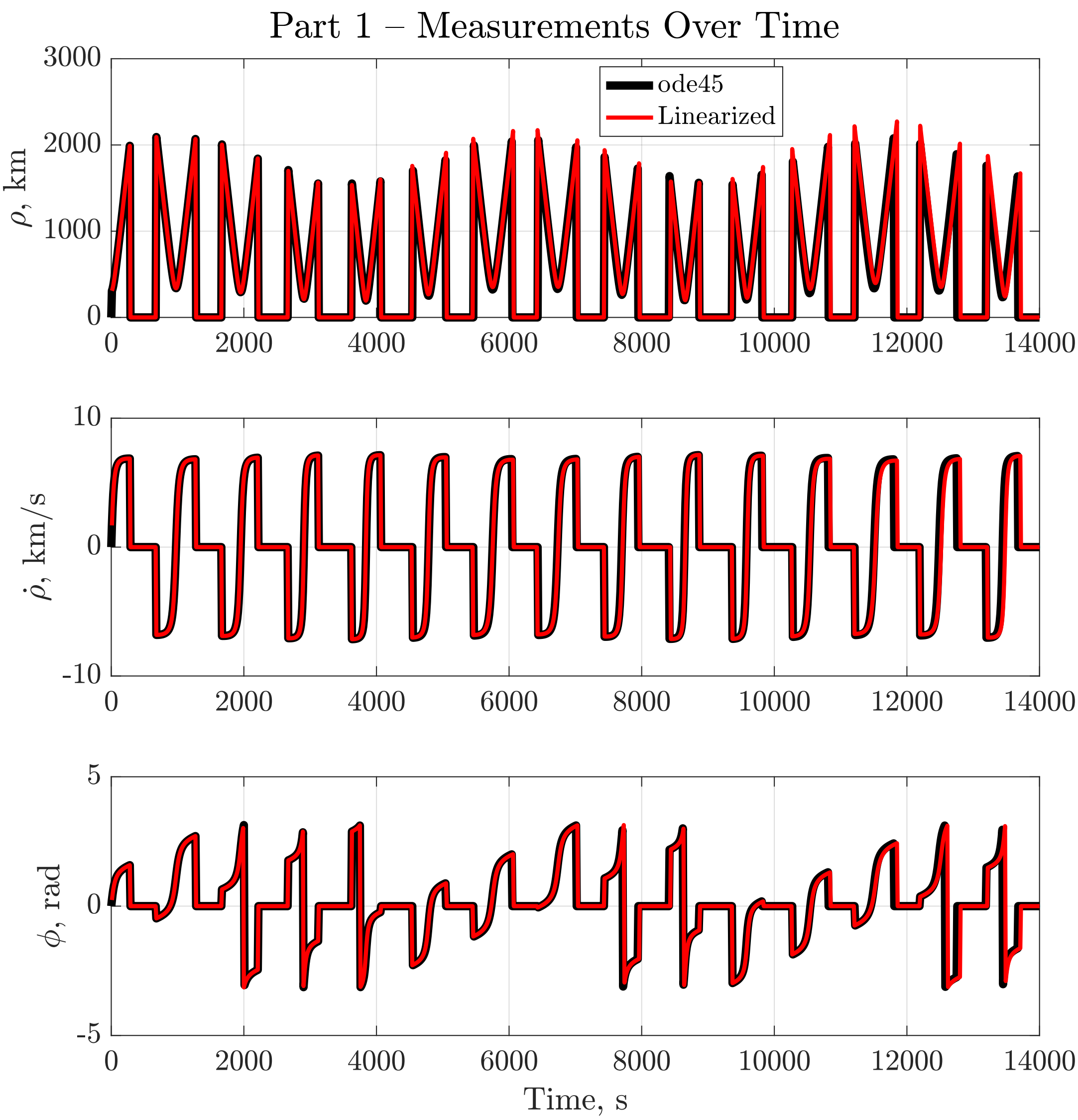
C. Simulating the Linearized Dynamics and Measurements

The plots below compare the results between the full non-linear dynamics and the DT linearized dynamics. As is shown below, the states and measurements match relatively well over time, but the residuals begin to grow as time goes on. The initial perturbation that was used for this analysis was $\Delta x = [0, 0.075, 0, -0.021]^T$.




States Over Time
×10^4
1
0
-1
x, km
ode45
Linearized
10
0
-10
ẋ, km/s
y, km
ẏ, km/s
0
2000
4000
6000
8000
10000
12000
14000
Time, s


Part 1 – State Perturbations
δ_x, km
ode45
Linearized
$\delta_{\dot{x}}$, km/s
δ_y, km
$\delta_{\dot{y}}$, km/s
Time, s



II. Linearized Kalman Filter

With a working linear model of the system, the next step is to introduce process and measurement noise and try to create a working filter. To test the filter, truth model testing (TMT) was used. Noisy states and measurements were simulated and passed into the Linearized Kalman Filter (LKF) and the output was compared with the input to determine if the filter was working properly. These simulations were run for 3 orbital periods. This number was chosen to get a good sample of data and to determine not only if the filter would work initially, but after awhile as well. If only one orbit was simulated, the filter could easily break online after the first orbit. A typical simulation is shown below in Figures 1 - 3. It is easy to see from Figures 1 and 2 the LKF does an okay job of approximating the orbit for most of the time. Depending on the particular simulation and the random noise added to the truth data, this can look better or worse. Each instance of the LKF was given the same initial conditions. These are the initial conditions for the nominal orbit (given above) along with the perturbation $\Delta x = [0, 0.075, 0, -0.021]^T$. The initial state error covariance matrix (P_0) is shown below:

$$P_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix} \tag{1}$$

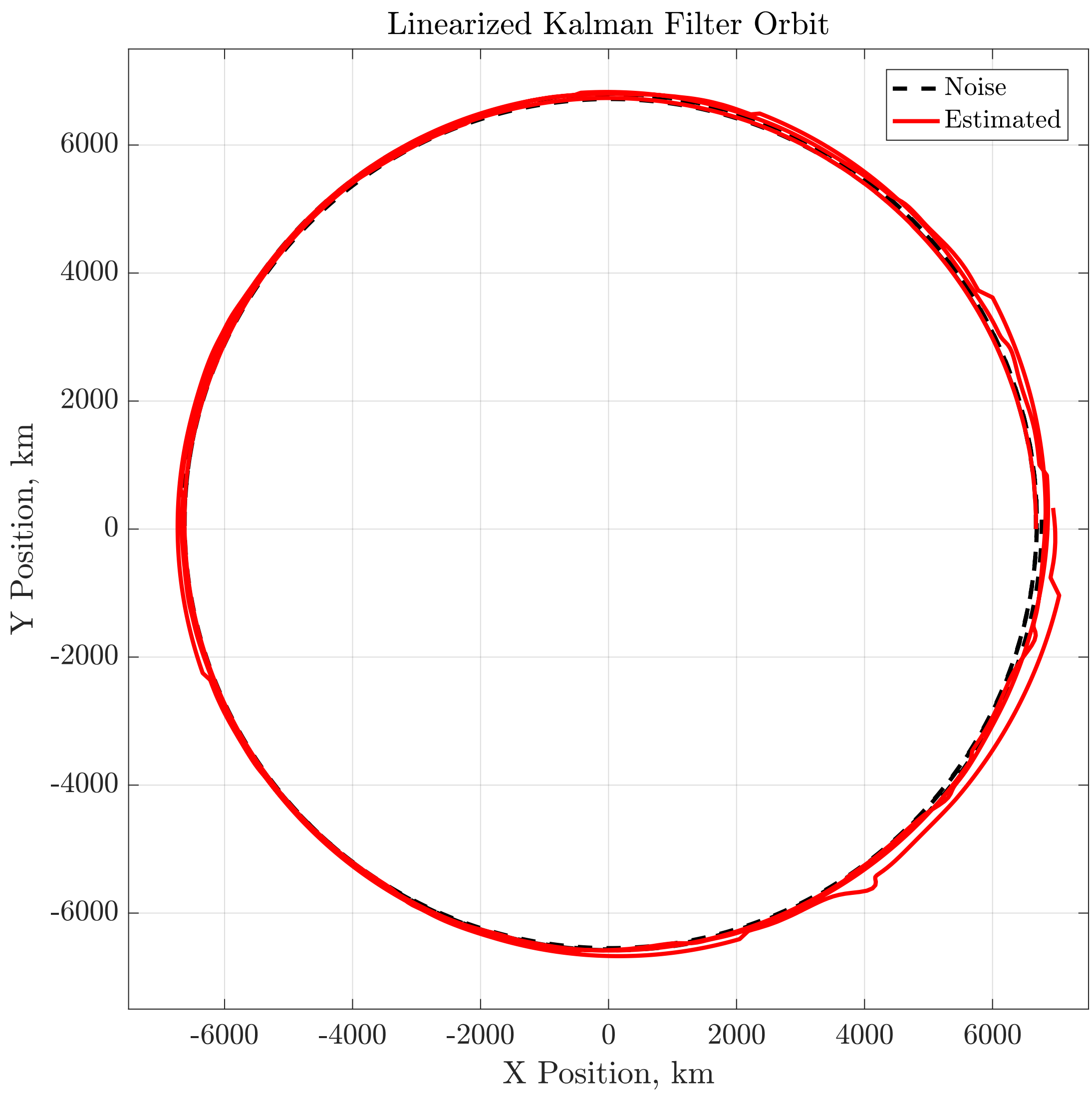


Figure 1: Linearized Kalman Filter Orbit

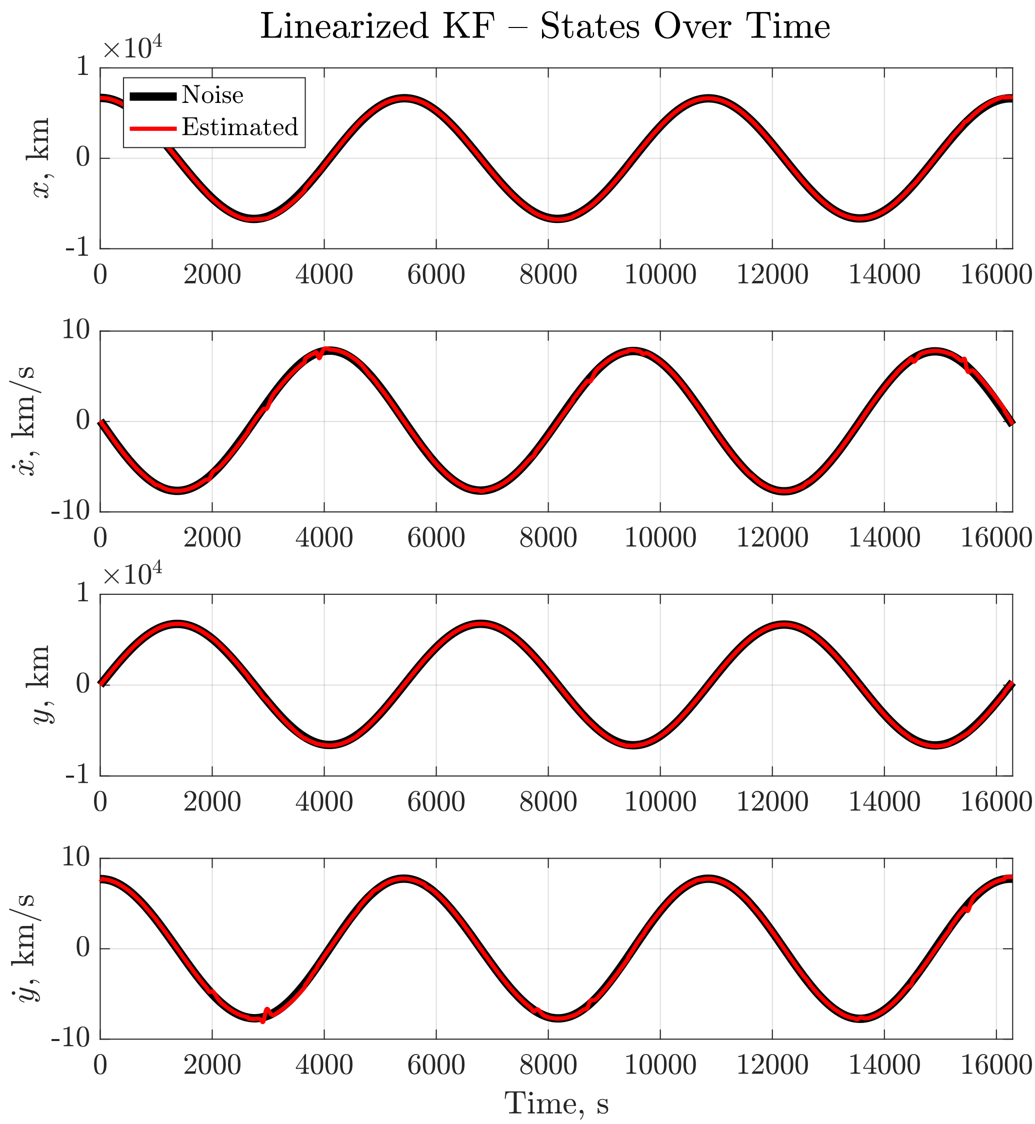


Figure 2: Linear Kalman Filter States over Time

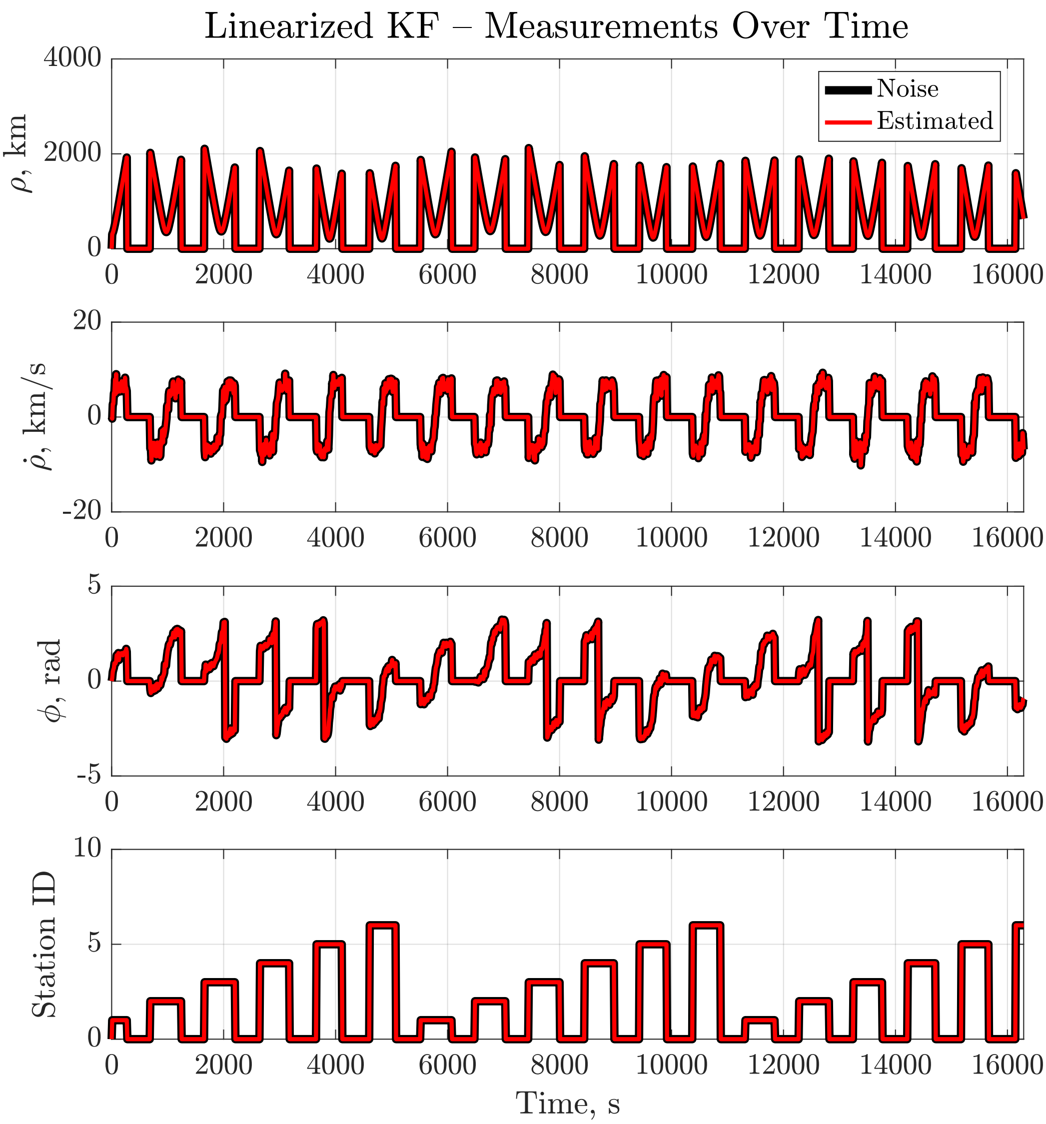


Figure 3: Linear Kalman Filter Measurements over Time

To figure out exactly what is going on here, we can look at how the state residuals evolve over time. The residuals are the difference between the estimated state and the actual state (simulated here with process and measurement noise for use to tune the filter and use TMT to determine if the filter is working). A typical instance of the state residuals is shown in Figure 4. It is easy to see the residuals often come out of the predicted 2σ bounds. As time goes on, the filter gets worse and begins to spend more and more time outside these bounds because the initial perturbation of the orbit and the noise are causing the actual orbit to get further away from the nominal orbit. Since the LKF bases its estimates on the nominal trajectory, the further away the true trajectory is, the more likely the filter is to break. Sometimes the filter recovers and returns inside the bounds and spends a little time successfully predicting the state of the satellite. Comparing Figures 2 and 3 shows the filter breaks the most when the satellite is out of range of all the ground stations, meaning it is receiving no new measurements and therefore using only prediction to estimate the state of the satellite.

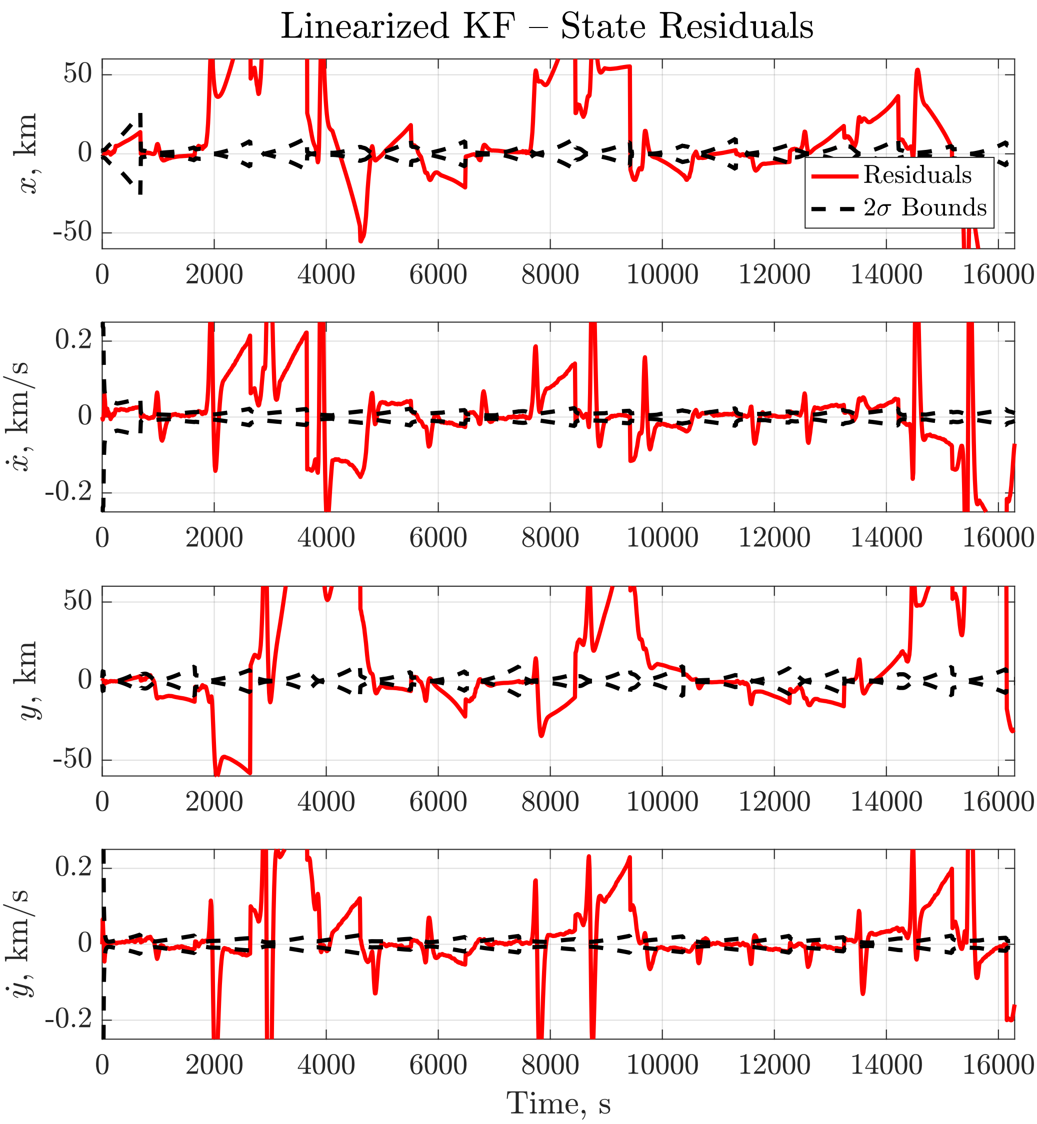


Figure 4: Linear Kalman Filter State Residuals over Time

Another way to check if an LKF is working is by using NEES and NIS testing. Due to the complexity of the measurements of this problem and the fact that measurements aren't always available, NIS testing was determined to be unnecessary for the purposes of this project However, an example of a NIS test will be provided for the EKF. For NEES testing, an α of 0.05 was selected. This gives the filter a 95% confidence level, meaning that it is acceptable for 5% of the NEES statistic points to fall outside of the bounds determined by the 95% confidence level. This value of α was chosen because a 95% confidence interval is relatively standard in statistics.

In a Linearized Kalman Filter, the filter predicts the deviation from the nominal trajectory (δx). The NEES statistic is defined in Equation 2:

$$\bar{\epsilon}_x = \vec{e}^T P^{-1} \vec{e} \tag{2}$$

where $\vec{e}$ is the error between the filter's prediction of the full state and the true full state. The filter's prediction of the full state is

$$\vec{x}_{LKF} = \vec{x}_{nominal} + \vec{\delta x} \tag{3}$$

where $\vec{\delta x}$ is what the LKF is actually predicting. This means the NEES statistic for the LKF is given by

$$\bar{\epsilon}_x = (\vec{x}_{true} - \vec{x}_{LKF})^T P^{-1} (\vec{x}_{true} - \vec{x}_{LKF}) = (\vec{x}_{true} - (\vec{x}_{nominal} + \delta\hat{x}))^T P^{-1} (\vec{x}_{true} - (\vec{x}_{nominal} + \delta\hat{x})) \tag{4}$$

In Equation 4, P is the state estimation error covariance matrix (produced and used by the LKF), $\vec{\delta x}$ is what the LKF is actually predicting, $\vec{x}_{nominal}$ is the nominal trajectory (a circular orbit in this case), and $\vec{x}_{true}$ is the true state of the spacecraft. The true state is only known because it is simulated. In reality, the true state would not be known. It is only used to determine if the filter is working in simulations before the filter is actually used.

The results of NEES testing are shown below in Figures 5 and 6. Figure 5 shows just how badly the LKF breaks as time goes on, especially when the satellite loses a measurement. Figure 6 shows the LKF performing well for the beginning of the simulation. This is when the satellite is still close to the nominal trajectory (noise and perturbations haven't changed the orbit too much at this point) and the satellite still has a measurement. As the satellite travels further away from the nominal orbit, the LKF becomes more likely to break (and eventually does) and the errors it produces become greater.

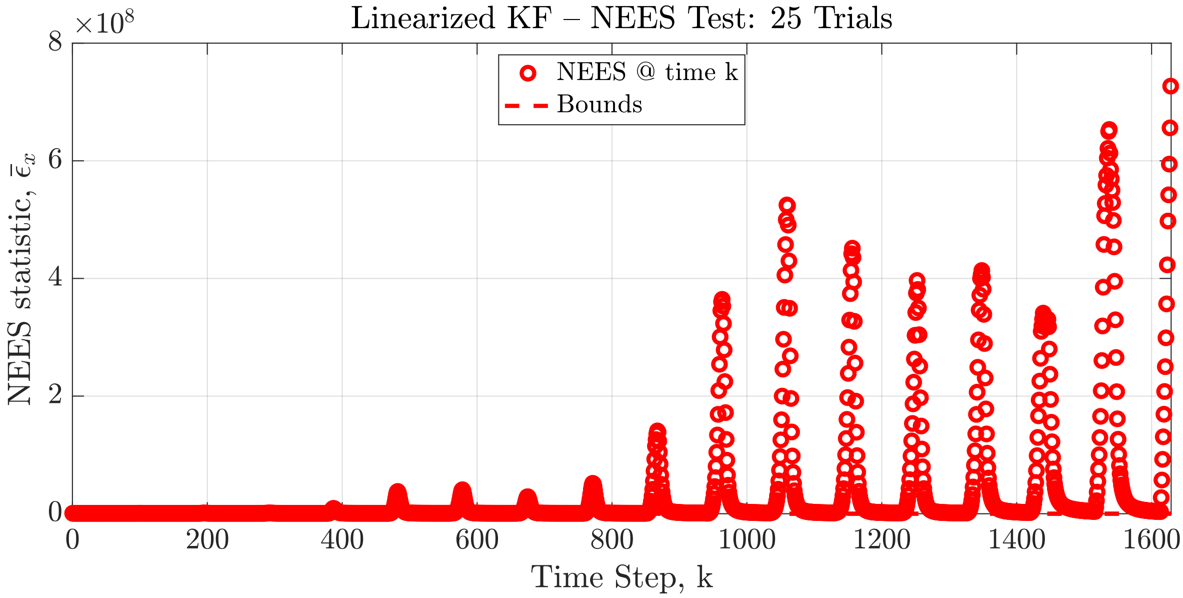


Figure 5: NEES Testing for LKF

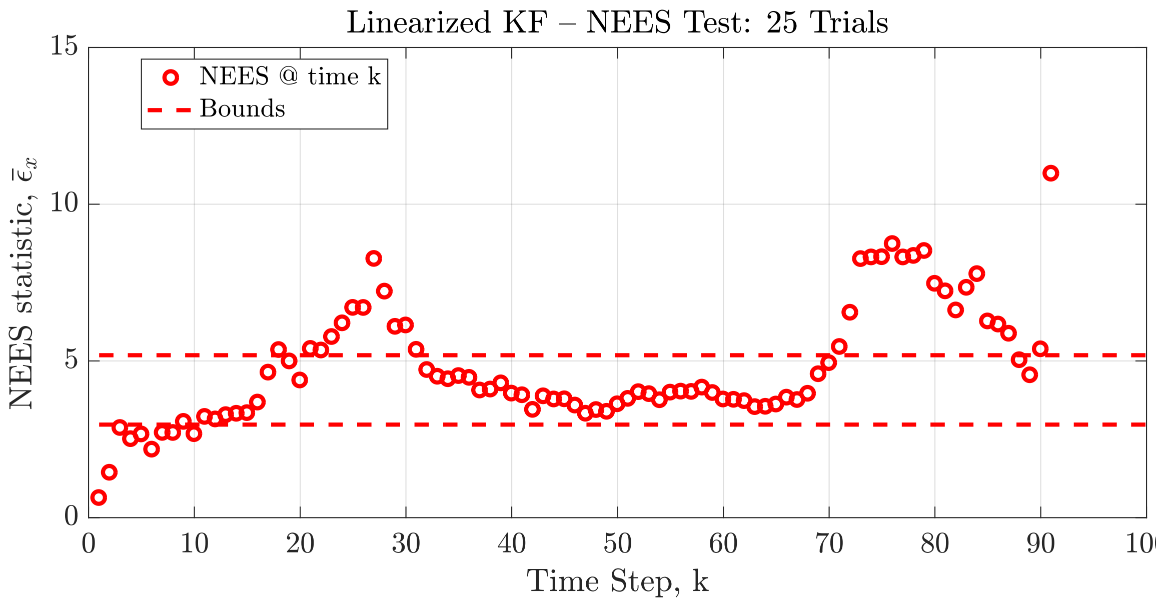


Figure 6: NEES Testing for LKF Zoomed View

Because the LKF breaks in such an extreme fashion, it is impossible to tune it perfectly. It was therefore left untuned because the NEES test falls within the bounds of being acceptable for the first time steps.

III. Extended Kalman Filter

Next is the Extended Kalman Filter (EKF). A typical result from the EKF is shown below in Figures 7 - 9. Again, 3 orbital periods were simulated to ensure the filter would properly work for longer durations. It is immediately clear when looking at these plots that the EKF does a much better job than the LKF tracking the satellite's true state when there is an initial perturbation and process and measurement noise to worry about.

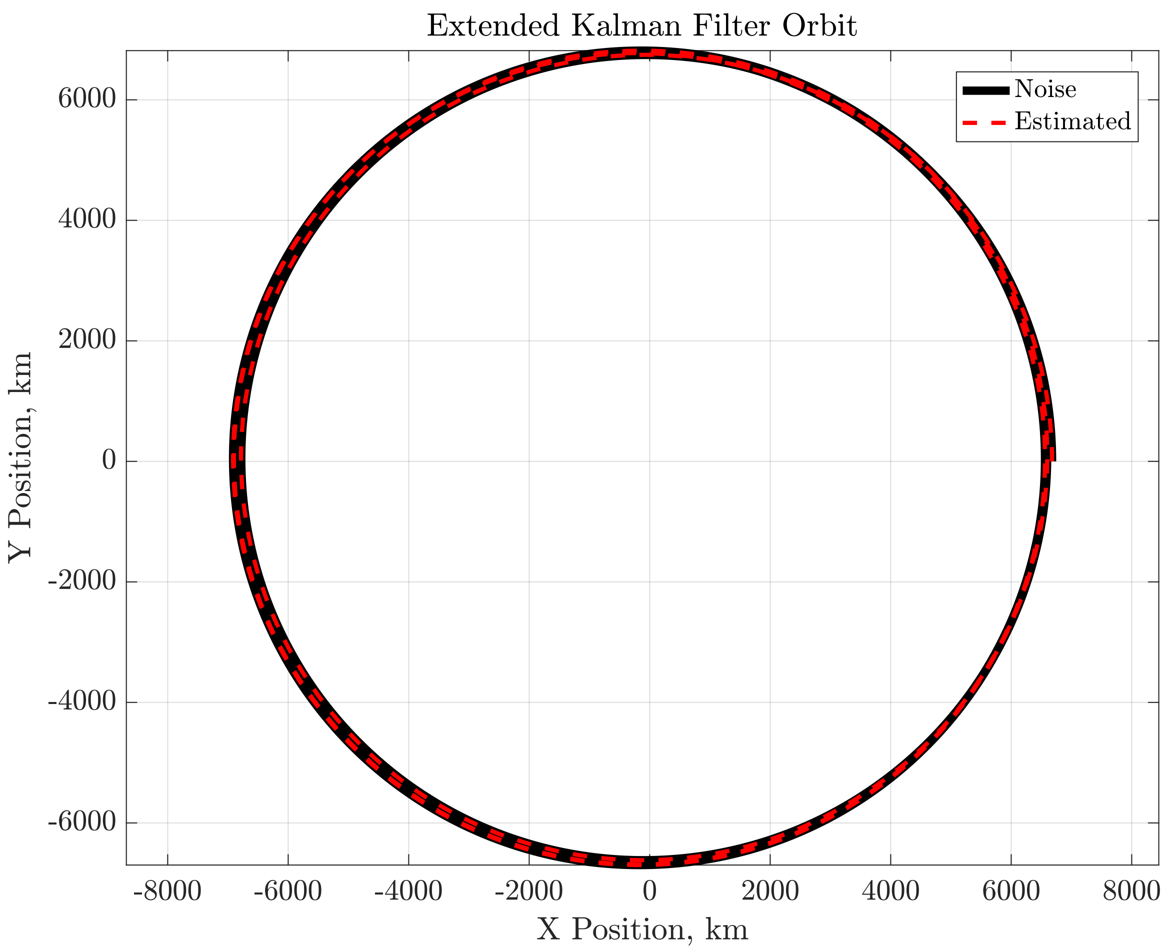


Figure 7: Extended Kalman Filter Orbit

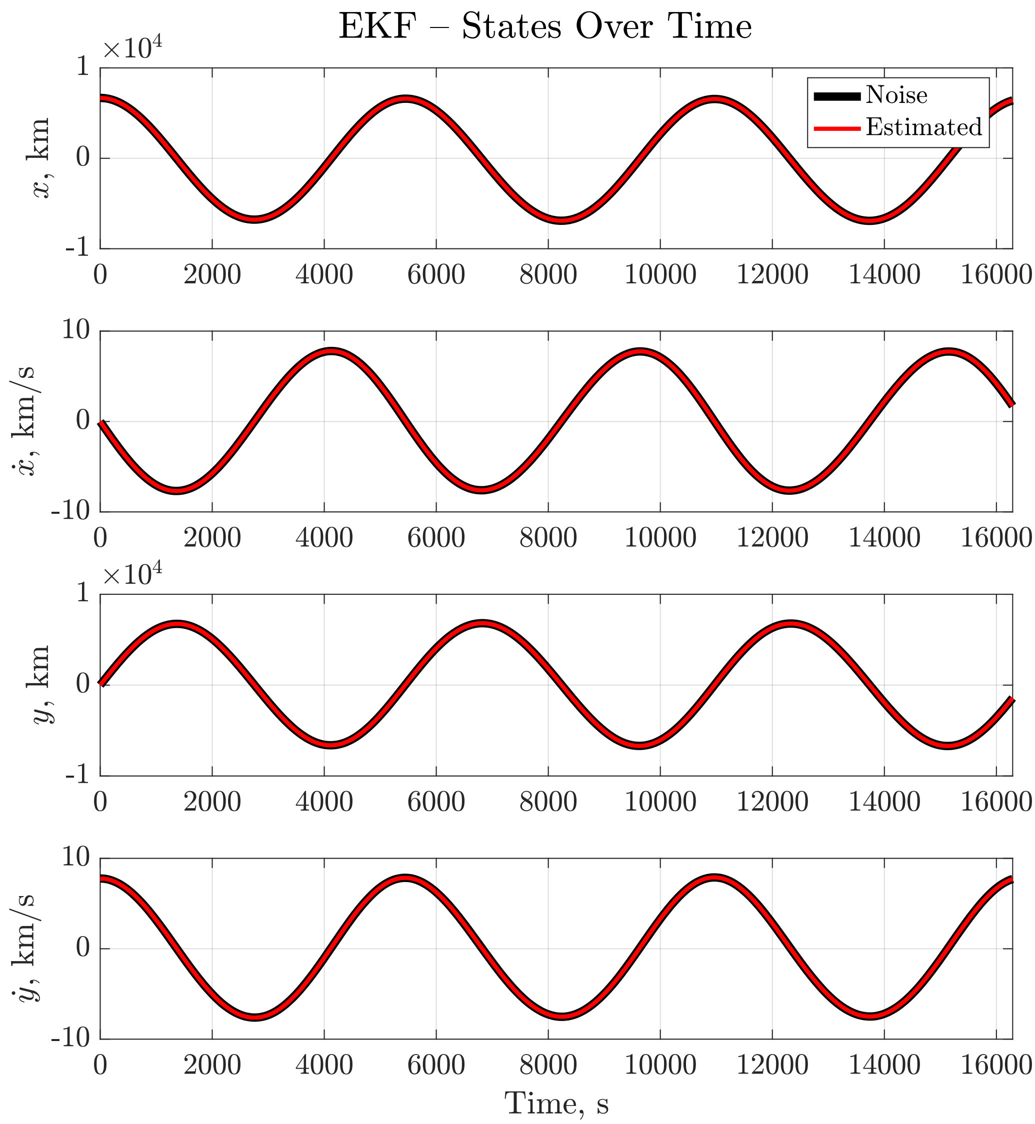


Figure 8: Extended Kalman Filter States

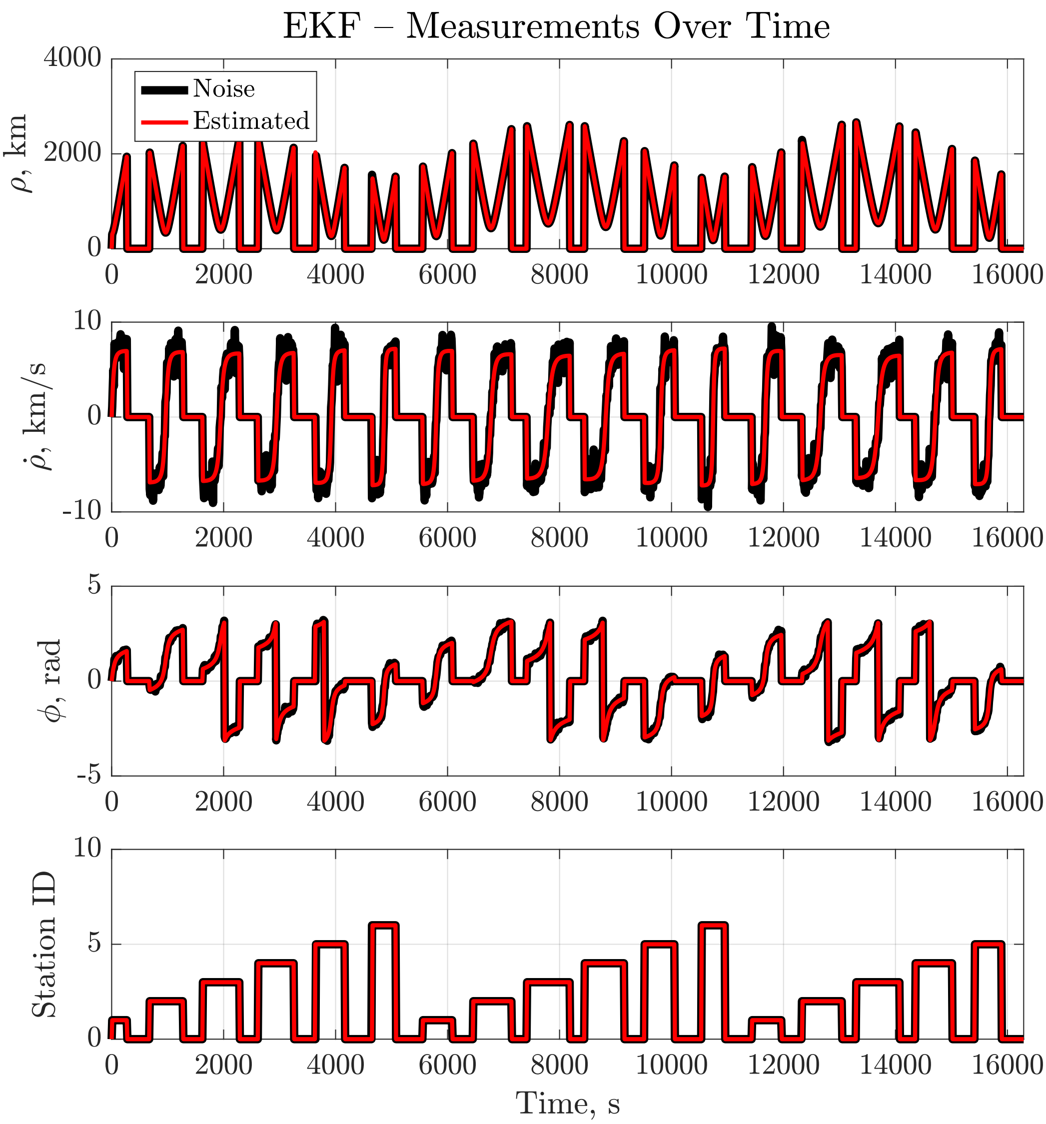


Figure 9: Extended Kalman Filter Measurements

Figure 10 shows a typical instance of the error between the EKF state and the true state over time along with the predicted 2σ bounds around that error. The error is almost always within the bounds, meaning the filter is working effectively. The state residual can go outside the bound because the bound is only 2σ, the residual should stay in that bound 95% of the time (which it does).

The initial state error covariance was very important when tuning the EKF. The standard deviation of the position should be 100 times the standard deviation of the velocity. This is because the time step is 10 seconds and velocity is approximately proportional to dt*position. Therefore, the initial covariance was set such that the initial position variance was 100 (dt^2) times the initial velocity variance because the process noise affects the states through $\tilde{\Omega} \approx dt * \Gamma$. This means the process noise is scaled by dt^2. This is shown below:

$$\hat{x}(k+1) = \tilde{F}(k)\hat{x}(k) + \tilde{G}(k)\vec{u}(k) + \tilde{\Omega}Q\tilde{\Omega}^T \tag{5}$$

$$P_0 = \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.001 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0.001 \end{bmatrix} \tag{6}$$

To tune the EKF, the NEES statistic, as well as the state residuals were examined. If the state residuals go outside of the 2σ bounds more than 95% of the time, the filter is not working properly. When the majority of NEES points were above the bounds, this meant there was too much noise in the filter, meaning Q was too big. If the majority of NEES points were below the bounds, there was not enough noise in the filter, and it became over confident. To fix this issue, the magnitude of Q could be increased. Because the true process noise was known, it was used as a starting point for tuning and refinements were made from there. The final Q used by the EKF (after tuning) was $0.95 * Q_{true}$ (Where Q_{true} is the process noise covariance used to simulate the actual process).

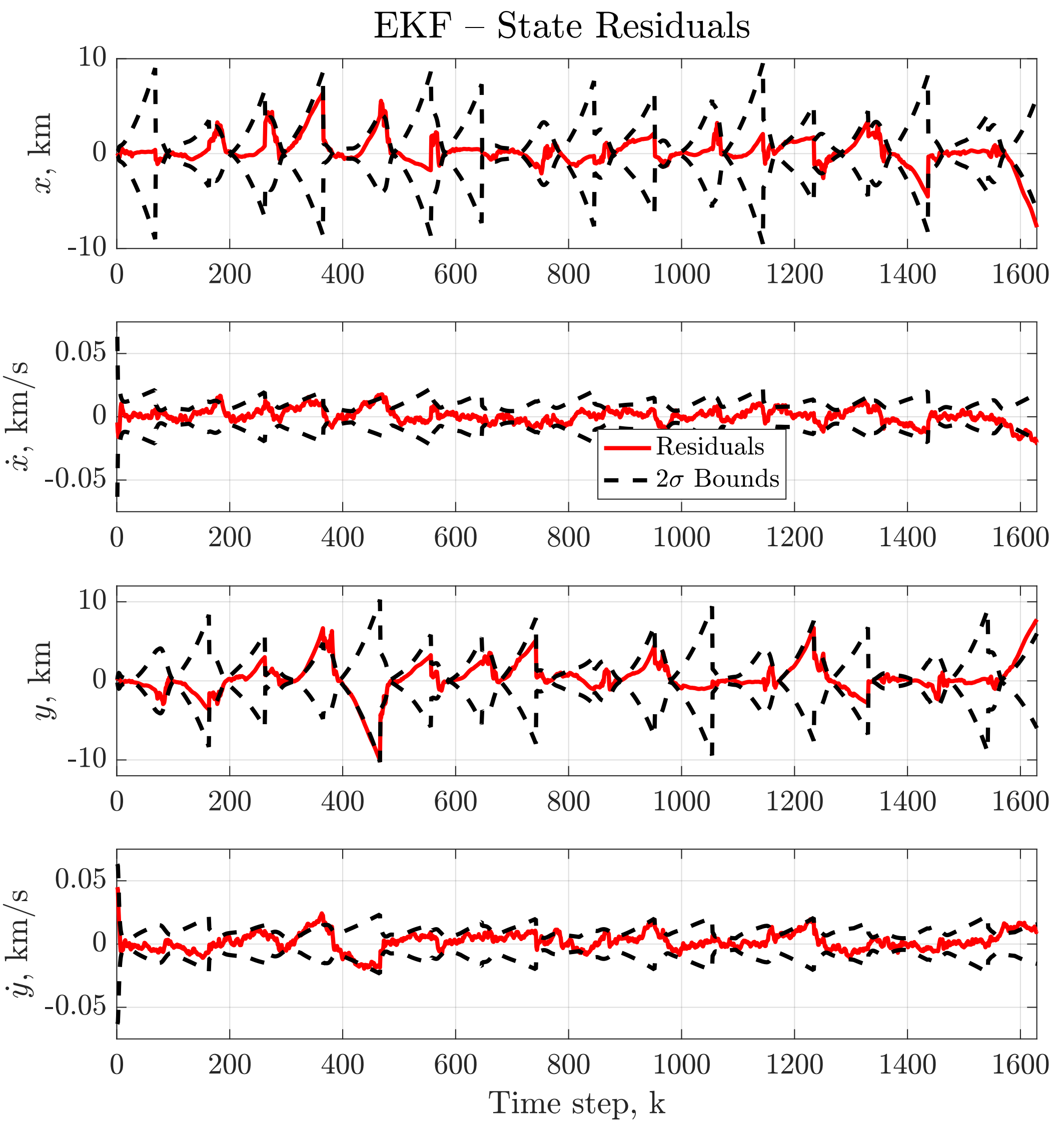


Figure 10: Extended Kalman Filter State Residuals

When calculating NEES for the EKF, the same α of 0.05 was used as in the LKF. This was chosen for the reasons explained above in the LKF section as well as to keep consistency for comparison purposes. The NEES statistic for the EKF is essentially the same as the NEES statistic for the LKF, but the details are slightly different. Where the LKF predicts the deviation from the nominal state, the EKF predicts the total state. This makes the equations slightly simpler. The general equation for NEES remains the same (Equation 2. The main difference is the definition of the error.

$$\vec{e} = \vec{x}_{true} - \hat{x} \tag{7}$$

where $\hat{x}$ is what the EKF is predicting. The full equation for the NEES statistic for the EKF is then:

$$\bar{\epsilon}_x = (\vec{x}_{true} - \hat{x})^T P^{-1} (\vec{x}_{true} - \hat{x}) \tag{8}$$

It is clear from Figure 11 that the filter is working properly as most of the points fall within the 95% confidence interval.

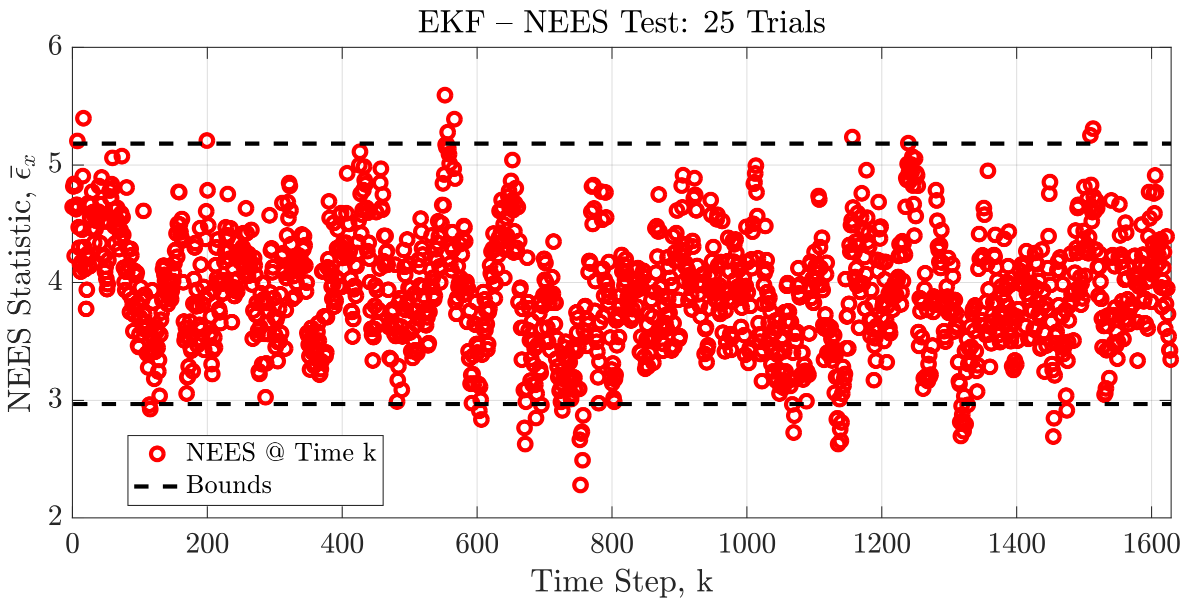


Figure 11: Extended Kalman NEES Testing

The NIS statistic has a slightly different form than the NEES statistic. The equation is shown below:

$$\bar{\epsilon}_y = \vec{e}_y^t P^{-1} \vec{e}_y \tag{9}$$

where $\vec{e}_y$ is the difference between the true measurement (in this case, the simulated nonlinear model with process and measurement noise) and the estimated measurement produced by the EKF. So the full equation for the NIS statistic is:

$$\bar{\epsilon}_y = (\vec{y}_{true} - \hat{y})^t P^{-1} (\vec{y}_{true} - \hat{y}) \tag{10}$$

where $\hat{y}$ is the measurement predicted by the EKF.

For this problem, there are not always measurements.When adding process and measurement noise to the simulated truth data, there can be a misalignment between when each simulation has a measurement. Therefore, it was important to use the MATLAB function *nanmean.m*, which is able to take the mean of all values that are not NaN. NaNs occur in the EKF when there is no station tracking the satellite on either the noisy or estimated data. When running multiple simulations, NaNs can appear within the same time step k for different simulations, which need to be removed. Figure 12 shows the results of the EKF NIS test after implementing *nanmean.m*. It is clear to see that the majority of points are within the bounds, which means that the EKF filter is working properly.

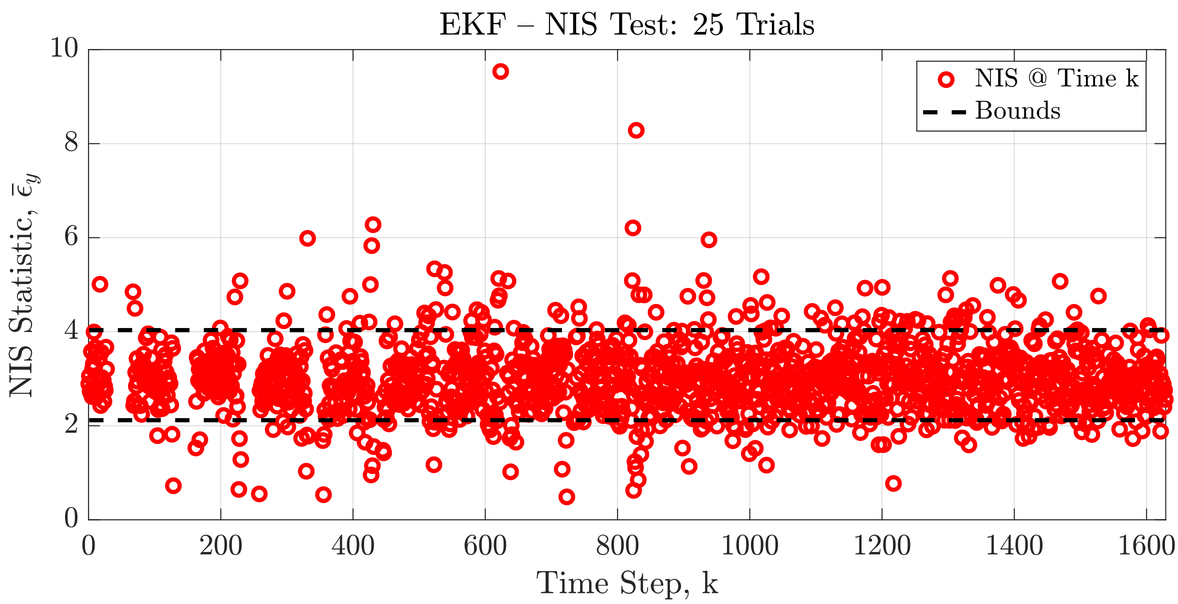


Figure 12: Extended Kalman NIS Testing

IV. Filter Testing

The following figures show how the tuned filters perform when exposed to actual data. From these figures (13 and 14), it is clear that the extended Kalman Filter does a better job at predicting the state of the satellite. The Linearized Kalman Filter has some very large spikes as time goes on. These spikes are incredibly unlikely to physically occur with an actual satellite. States produced by the Extended Kalman Filter, on the other hand, are wonderfully smooth and seem to predict a more physically possible orbit. The two filters seem to agree early on in the orbit, but the linear filter loses fidelity as time goes on. This is not surprising given the earlier results with NEES testing on both filters. The EKF performed well for multiple orbits, while the LKF struggled to perform well in NEES testing for the first 100 time steps (1000 seconds).

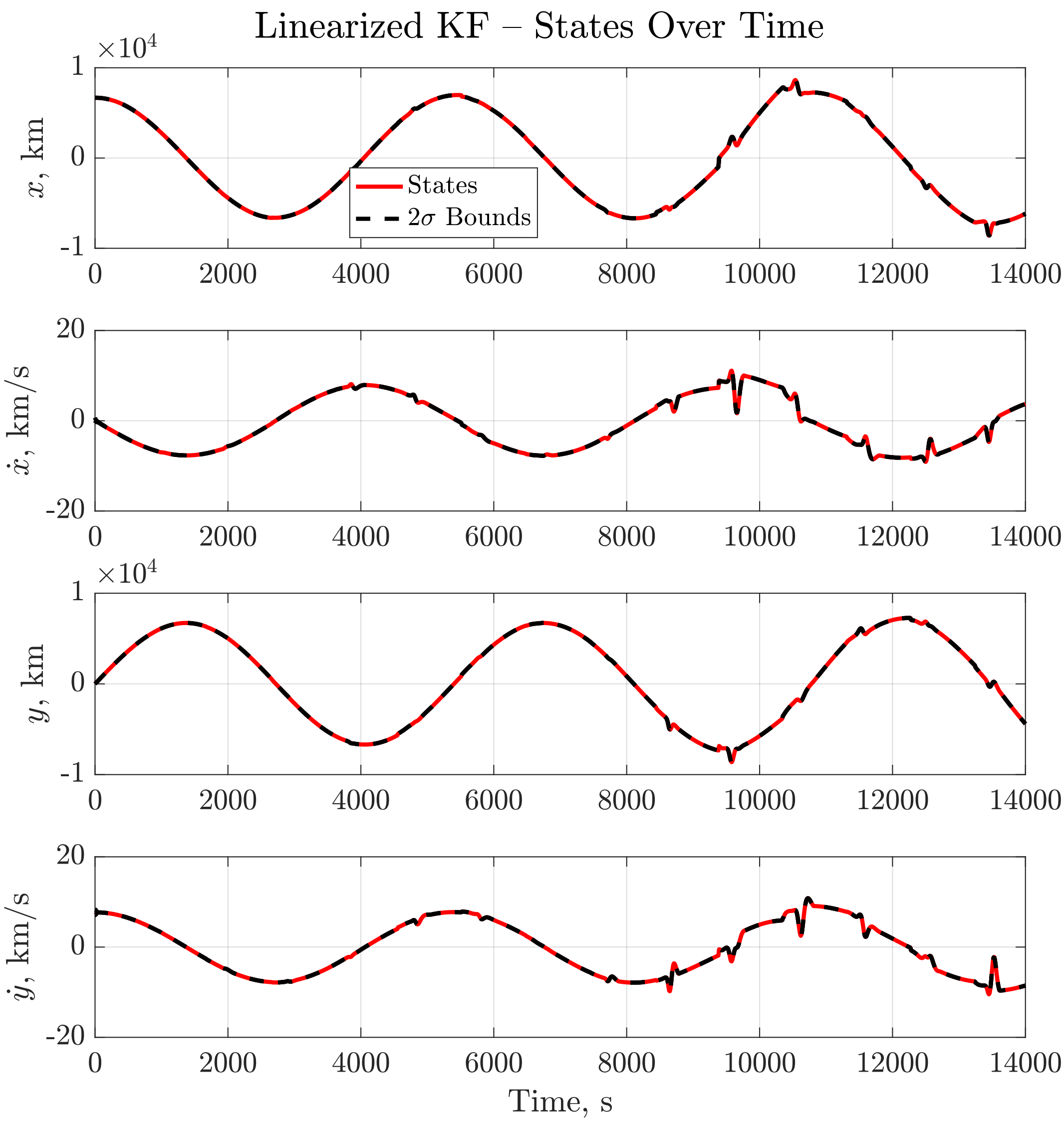


Figure 13: Linearized KF using true measurements

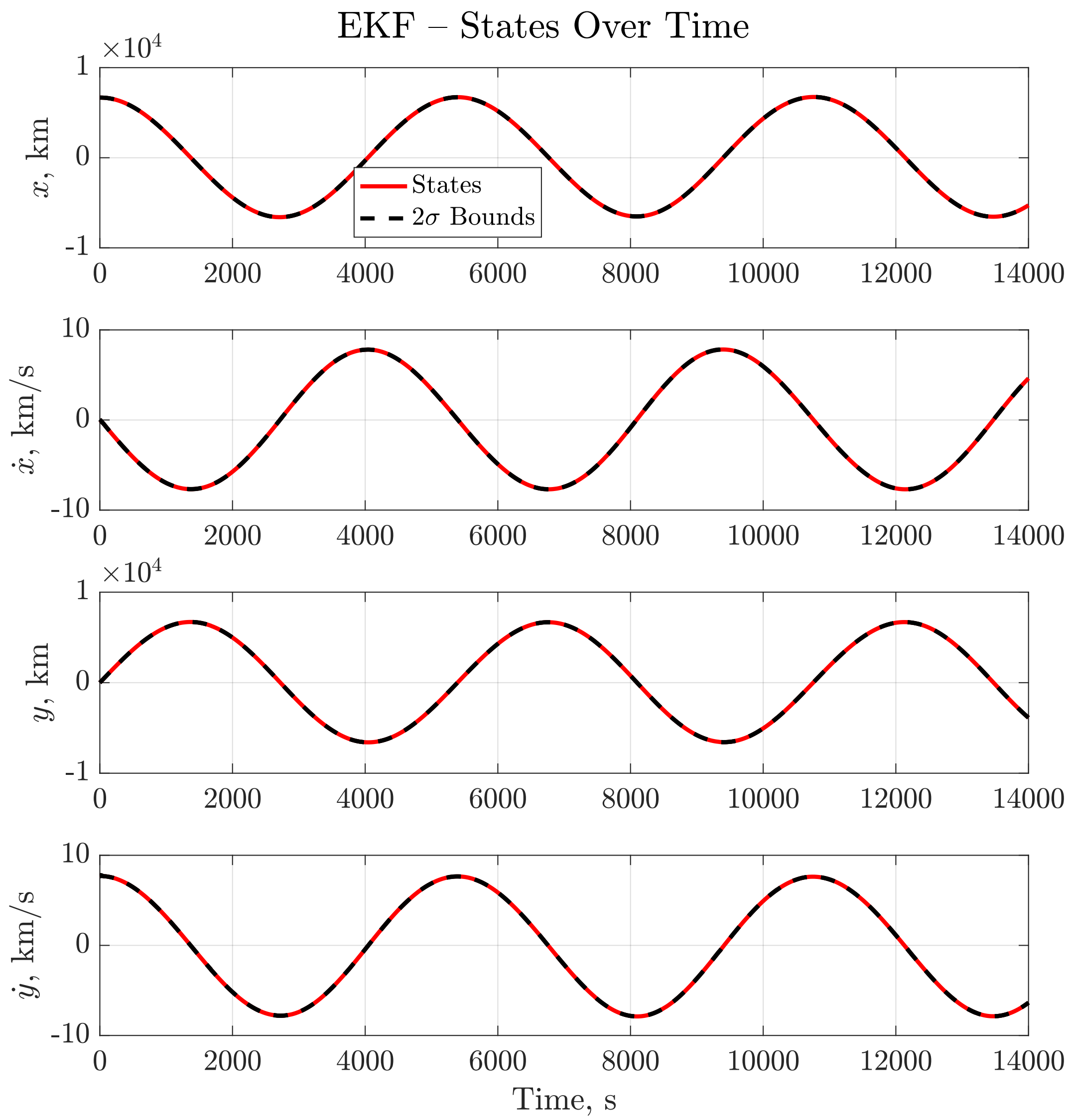


Figure 14: Extended KF using true measurements

V. Haiku

Educated guess?
Estimation is much more
Just talk to Kalman

*Orbit Determination

2 $r = \sqrt{x^2+y^2}$ $\qquad \mu = 398600 \text{ km}^3/s^2$

$$\ddot{x} = -\frac{\mu x}{r^3} + u_1 + \tilde{w}_1$$

$$\ddot{y} = -\frac{\mu y}{r^3} + u_2 + \tilde{w}_2$$

circular orbit, $h = 300\text{km} \rightarrow r_0 = 6678 \text{ km}$

$x_0 = 6678 \text{ km}$, $y_0 = 0 \text{ km}$
$\dot{x}_0 = 0 \text{ km/s}$, $\dot{y}_0 = r_0\sqrt{\frac{\mu}{r_0^3}}$

$$x = \begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \end{bmatrix} \quad u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad \tilde{w} = \begin{bmatrix} \tilde{w}_1 \\ \tilde{w}_2 \end{bmatrix}$$

a) Find Jacobian to obtain CT LTI model parameters

$$\left[\frac{\partial f}{\partial x}\right]_{nom} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{nom}$$

$$x = \begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \rightarrow \dot{x} = \begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{y} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} \text{ or } \begin{bmatrix} x_2 \\ \dot{x}_2 \\ x_4 \\ \dot{x}_4 \end{bmatrix}$$

$$\ddot{x} = f_2 = -\frac{\mu x_1}{(x_1^2+x_3^2)^{3/2}} + u_1 + \tilde{w}_1 \qquad f_1 = \dot{x} = x_2, \quad f_3 = \dot{y} = x_4$$

$$\ddot{y} = f_4 = -\frac{\mu x_3}{(x_1^2+x_3^2)^{3/2}} + u_2 + \tilde{w}_2$$

$$\frac{\partial f_1}{\partial x_1} = 0, \quad \frac{\partial f_1}{\partial x_2} = 1, \quad \frac{\partial f_1}{\partial x_3} = 0, \quad \frac{\partial f_1}{\partial x_4} = 0$$

$$\frac{\partial f_3}{\partial x_1} = 0, \quad \frac{\partial f_3}{\partial x_2} = 0, \quad \frac{\partial f_3}{\partial x_3} = 0, \quad \frac{\partial f_3}{\partial x_4} = 1$$

2 a)

$$\frac{\partial f_2}{\partial x_1} = \frac{\partial}{\partial x_1}\left[\frac{-\mu x_1}{(x_1^2+x_3^2)^{3/2}} + u_1 + \tilde{w}_1\right] \rightarrow -\mu \frac{\partial}{\partial x_1}\left[x_1(x_1^2+x_3^2)^{-3/2}\right]$$

$$= -\mu\left[x_1\left(-\frac{3}{2}(x_1^2+x_3^2)^{-5/2}(2x_1)\right) + (x_1^2+x_3^2)^{-3/2}(1)\right]$$

$$= -\mu\left[\frac{-3x_1^2}{(x_1^2+x_3^2)^{5/2}} + \frac{1}{(x_1^2+x_3^2)^{3/2}}\right]$$

$$= -\mu\left[\frac{-3x_1^2+x_1^2+x_3^2}{(x_1^2+x_3^2)^{5/2}}\right]$$

$$= \mu\left[\frac{2x_1^2-x_3^2}{r^5}\right]$$

$$\frac{\partial f_2}{\partial x_3} = \frac{\partial}{\partial x_2}\left[\frac{-\mu x_1}{(x_1^2+x_3^2)^{3/2}} + u_1 + \tilde{w}_1\right] \rightarrow -\mu x_1 \frac{\partial}{\partial x_2}\left[(x_1^2+x_3^2)^{-3/2}\right]$$

$$= -\mu x_1\left[-\frac{3}{2}(x_1^2+x_3^2)^{-5/2}(2x_3)\right]$$

$$= -\mu x_1\left[\frac{-3x_3}{(x_1^2+x_3^2)^{5/2}}\right]$$

$$= \frac{3\mu x_1 x_3}{r^5}$$

$$\frac{\partial f_2}{\partial x_2} = 0 \quad , \quad \frac{\partial f_2}{\partial x_4} = 0$$

$$\frac{\partial f_4}{\partial x_1} = -\mu x_3 \frac{\partial}{\partial x_1}\left[(x_1^2+x_3^2)^{-3/2}\right] = -\mu x_3\left[-\frac{3}{2}(x_1^2+x_3^2)^{-5/2}\right](2x_1)$$

$$= -\mu x_3\left[\frac{-3x_1}{r^5}\right] = \frac{3\mu x_1 x_3}{r^5}$$

$$\frac{\partial f_4}{\partial x_3} = -\mu \frac{\partial}{\partial x_3}\left[x_3(x_1^2+x_3^2)^{-3/2}\right] = -\mu\left[x_3\left(-\frac{3}{2}(x_1^2+x_3^2)^{-5/2}(2x_3)\right) + (x_1^2+x_3^2)^{-3/2}(1)\right]$$

$$= -\mu\left[\frac{-3x_3^2}{r^5} + \frac{1}{r^3}\right]$$

$$= -\mu\left[\frac{-3x_3^2+x_1^2+x_3^2}{r^5}\right] = \mu\left[\frac{2x_3^2-x_1^2}{r^5}\right]$$

$$\frac{\partial f_4}{\partial x_2} = 0 \quad , \quad \frac{\partial f_4}{\partial x_4} = 0$$

Mitchell Smith
Chris Chamberlain

7/10

2 a)

$$\boxed{A_{nom} = \left[\frac{\partial f}{\partial x}\right]_{nom} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \mu\left[\frac{2x_1^2 - x_3^2}{r^5}\right] & 0 & \frac{3\mu x_1 x_3}{r^5} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{3\mu x_1 x_3}{r^5} & 0 & \mu\left[\frac{2x_3^2 - x_1^2}{r^5}\right] & 0 \end{bmatrix}_{nom}}$$

A_{nom} will have dimensions of (4×4) since there are 4 states

$$B_{nom} = \left[\frac{\partial f}{\partial u}\right]_{nom} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} \\ \vdots & \vdots \\ \frac{\partial f_4}{\partial u_1} & \frac{\partial f_4}{\partial u_2} \end{bmatrix}$$

$$\frac{\partial f_1}{\partial u_1} = 0, \quad \frac{\partial f_1}{\partial u_2} = 0, \quad \frac{\partial f_3}{\partial u_1} = 0, \quad \frac{\partial f_3}{\partial u_2} = 0$$

$$\frac{\partial f_2}{\partial u_1} = 1, \quad \frac{\partial f_2}{\partial u_2} = 0, \quad \frac{\partial f_4}{\partial u_1} = 0, \quad \frac{\partial f_4}{\partial u_2} = 1$$

$$\boxed{B_{nom} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}_{nom}}$$

B_{nom} has dimensions of (4×2) since there are 4 states and 2 inputs

2 a) $y^i(t) = \begin{bmatrix} \rho^i(t) \\ \dot{\rho}^i(t) \\ \phi^i(t) \end{bmatrix} + \tilde{v}^i(t)$ $\qquad X = x_1, \quad Y = x_3$; $\dot{X} = x_2, \quad \dot{Y} = x_4$

$$h_1 = \rho^i(t) = \sqrt{(x_1 - x_s)^2 + (x_3 - y_s)^2}$$

$$h_2 = \dot{\rho}^i(t) = \frac{[x_1 - x_s][x_2 - \dot{x}_s] + [x_3 - y_s][x_4 - \dot{y}_s]}{[(x_1 - x_s)^2 + (x_3 - y_s)^2]^{1/2}}$$

$$h_3 = \phi^i(t) = \tan^{-1}\left(\frac{x_3 - y_s}{x_1 - x_s}\right)$$

$$C_{nom} = \left[\frac{\partial h}{\partial x}\Big|_{nom}\right]$$

$$* \quad \frac{\partial h_1}{\partial x_2} = 0, \quad \frac{\partial h_1}{\partial x_4} = 0$$

$$\frac{\partial h_1}{\partial x_1} = \frac{\partial}{\partial x_1}\left[\left((x_1 - x_s)^2 + (x_3 - y_s)^2\right)^{1/2}\right] = \frac{1}{2}\left((x_1 - x_s)^2 + (x_3 - y_s)^2\right)^{-1/2}(2(x_1 - x_s))$$

$$* \quad \frac{\partial h_1}{\partial x_1} = \frac{x_1 - x_s}{[(x_1 - x_s)^2 + (x_3 - y_s)^2]^{1/2}}$$

$$\frac{\partial h_1}{\partial x_3} = \frac{\partial}{\partial x_3}\left[\left((x_1 - x_s)^2 + (x_3 - y_s)^2\right)^{1/2}\right] = \frac{1}{2}\left((x_1 - x_s)^2 + (x_3 - y_s)^2\right)^{-1/2}(2(x_3 - y_s))$$

$$* \quad \frac{\partial h_1}{\partial x_3} = \frac{x_3 - y_s}{[(x_1 - x_s)^2 + (x_3 - y_s)^2]^{1/2}}$$

2 a) $$\frac{\partial h_2}{\partial x_1} = \frac{\partial}{\partial x}\left[\frac{[x_1-x_s][x_2-\dot{x}_s]+[x_3-y_s][x_4-\dot{y}_s]}{[(x_1-x_s)^2+(x_3-y_s)^2]^{1/2}}\right]$$

$$= \frac{\partial}{\partial x_1}\left[(x_1x_2 - x_1\dot{x}_s - x_2x_s + x_s\dot{x}_s + x_3x_4 - x_3\dot{y}_s - x_4y_s + y_s\dot{y}_s)\left((x_1-x_s)^2+(x_3-y_s)^2\right)^{-1/2}\right]$$

$$= \left[(x_1-x_s)(x_2-\dot{x}_s)+(x_3-y_s)(x_4-\dot{y}_s)\right]\left[-\frac{1}{2}\left((x_1-x_s)^2+(x_3-y_s)^2\right)^{-3/2}(2(x_1-x_s))\right] + \left((x_1-x_s)^2+(x_3-y_s)^2\right)^{-1/2}(x_2-\dot{x}_s)$$

* $$\frac{\partial h_2}{\partial x_1} = \frac{x_2-\dot{x}_s}{[(x_1-x_s)^2+(x_3-y_s)^2]^{1/2}} - \frac{(x_1-x_s)[(x_1-x_s)(x_2-\dot{x}_s)+(x_3-y_s)(x_4-\dot{y}_s)]}{[(x_1-x_s)^2+(x_3-y_s)^2]^{3/2}}$$

* $$\frac{\partial h_2}{\partial x_2} = \frac{x_1-x_s}{[(x_1-x_s)^2+(x_3-y_s)^2]^{1/2}}$$ using numerator expansion above

$$\frac{\partial h_2}{\partial x_3} = \frac{\partial}{\partial x_3}\left[(x_1x_2 - x_1\dot{x}_s - x_2x_s + x_s\dot{x}_s + x_3x_4 - x_3\dot{y}_s - x_4y_s + y_s\dot{y}_s)\left((x_1-x_s)^2+(x_3-y_s)^2\right)^{-1/2}\right]$$

$$= \left[(x_1-x_s)(x_2-\dot{x}_s)+(x_3-y_s)(x_4-\dot{y}_s)\right]\left[-\frac{1}{2}\left((x_1-x_s)^2+(x_3-y_s)^2\right)^{-3/2}(2(x_3-y_s))\right] + \left((x_1-x_s)^2+(x_3-y_s)^2\right)^{-1/2}(x_4-\dot{y}_s)$$

* $$\frac{\partial h_2}{\partial x_3} = \frac{x_4-\dot{y}_s}{[(x_1-x_s)^2+(x_3-y_s)^2]^{1/2}} - \frac{(x_3-y_s)[(x_1-x_s)(x_2-\dot{x}_s)+(x_3-y_s)(x_4-\dot{y}_s)]}{[(x_1-x_s)^2+(x_3-y_s)^2]^{3/2}}$$

* $$\frac{\partial h_2}{\partial x_4} = \frac{x_3-y_s}{[(x_1-x_s)^2+(x_3-y_s)^2]^{1/2}}$$ using numerator expansion above

* $$\frac{\partial h_3}{\partial x_2} = 0 \quad , \quad \frac{\partial h_3}{\partial x_4} = 0$$

* $$\frac{\partial h_3}{\partial x_1} = \frac{\partial}{\partial x_1}\left[\tan^{-1}\left(\frac{x_3-y_s}{x_1-x_s}\right)\right] = \frac{y_s-x_3}{(x_s-x_1)^2+(y_s-x_3)^2}$$

* $$\frac{\partial h_3}{\partial x_3} = \frac{\partial}{\partial x_3}\left[\tan^{-1}\left(\frac{x_3-y_s}{x_1-x_s}\right)\right] = \frac{x_1-x_s}{(x_s-x_1)^2+(y_s-x_3)^2}$$

All equations with an * will make up the C_{nom} matrix. There will be 6 of the matrices due to there being 6 tracking stations where $x_s, \dot{x}_s, y_s, \dot{y}_s$ will change depending on which station is tracking. One of these matrices will have dimension (3 x 4)

Due: 12/6/17 | ASEN 5044 HW 8 | Mitchell Smith, Chris Chamberlain | 10/10

2 b) Observability: $\mathcal{O} = \begin{bmatrix} H \\ HF \\ \vdots \\ HF^{n-1} \end{bmatrix}$, Controllability: $\mathcal{C} = [G \; FG \cdots F^{n-1}G]$

& Stability

$\tilde{F}_k = I + \Delta t \tilde{A}_{nom[k]}$, $\tilde{G}_k = \Delta t \tilde{B}_{nom[k]}$, $\tilde{\Omega}_k = \Delta t \Gamma(t = t_k)_{nom[k]}$

*Since $\tilde{A}$ is a time varying matrix, and the system does not have a static equilibrium,

2 c) *All plots have been attached*