ASEN 5044 Final Project Report:

Statistical Orbit Determination of Satellite in Circular Orbital Plane

John Glista
University of Colorado – Boulder, Boulder, CO, 80309, USA

Statistical estimation is a powerful tool that is utilized across many varying fields of industry, including finance, medicine, and engineering. In the field of satellite orbit determination, a series of ground-based tracking stations can be used to measure the range and range-rates of Earth-orbiting objects. This paper presents the implementation of different types of predictor-corrector estimators, more generally known as Kalman filters. Specifically, the Linearized Kalman Filter (LKF), Extended Kalman Filter (EKF), and Unscented Kalman Filter (UKF) will be examined, with their effectiveness compared.

I. Nomenclature

X, Y	=	Earth-centered coordinates of the satellite on the orbital plane [km]
r	=	distance of satellite from the center of the Earth [km]
μ	=	standard gravitational parameter [km^3/s^2]
u_1, u_2	=	control accelerations
$\tilde{w}_1, \tilde{w}_2$	=	system disturbances
r_0	=	nominal orbit radius [km]
$\boldsymbol{x}(t)$	=	state space vector form for full states
$\boldsymbol{u}(t)$	=	state space vector form for control accelerations
$\tilde{\boldsymbol{w}}(t)$	=	state space vector form for system disturbances
i	=	ground-based tracking station ID number
$\rho^i(t)$	=	range measurement for specified tracking station
$\dot{\rho}^i(t)$	=	range-rate measurement for specified tracking station
$\phi^i(t)$	=	range of possible angles in which a station can measure the satellite
R_E	=	radius of the Earth [km]
ω_E	=	angular velocity of the Earth [rad/s]
LKF	=	linearized Kalman filter
EKF	=	extended Kalman filter
UKF	=	unscented Kalman filter
NEES	=	normalized estimation error squared
NIS	=	normalized innovation squared
TMT	=	Monte Carlo truth model test
χ^2	=	chi-square distribution
$\tilde{A}, \tilde{B}, \tilde{C}, \Gamma$	=	continuous time (CT) system Jacobians
$\tilde{F}, \tilde{G}, \tilde{H}, \tilde{\Omega}$	=	discrete time (DT) system Jacobians

II. Introduction

Several methods can be used in order to keep track of objects orbiting the earth. A common system is the utilization of ground-based stations which measure the range and range-rates of the satellite of interest. With the assumption that the location of the ground-stations and rotational velocity of the earth is known, accurate measurements can be

estimated for the satellite's position and velocity states. While the states of the satellite can be estimated, the uncertainty of this estimate is an equally important to know.

The implementation of the predicator-corrector class of estimators known as Kalman filters are explored in this paper. First, the continuous and discrete Jacobians for the deterministic system are derived from the full nonlinear equations of motion. The DT Jacobians will be utilized in both the linearized Kalman filter (LKF) and extended Kalman filter (EKF) for prediction and measurement updates. The system process noise and sensor measurement noises will also be considered through their respective Q and R covariance matrices. Finally, these Kalman filters will be evaluated by using the normalized estimate error squared (NEES) and normalized innovation squared (NIS) to determine the most suitable process noise covariance matrices.

III. Section 1: Deterministic System Analysis

It is necessary to accurately define the system dynamics of the orbiting satellite. This is achieved by determining the appropriate state Jacobians. The full nonlinear equations of motion for the orbiting satellite is given by:

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

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

with the full set of state, input, and system disturbance, and measurement state-space vectors as:

$$\boldsymbol{x}(t) = \left[X, \dot{X}, Y, \dot{Y}\right]^T = [x_1, x_2, x_3, x_4]$$

$$\boldsymbol{u}(t) = [u_1, u_2]^T$$

$$\widetilde{\boldsymbol{w}}(t) = [\widetilde{w}_1, \widetilde{w}_2]^T$$

$$\boldsymbol{y}^i(t) = \begin{bmatrix} \rho^i(t) \\ \dot{\rho}^i(t) \\ \phi^i(t) \end{bmatrix} + \tilde{v}^i(t)$$

A. Continuous-Time Jacobians

The standard nonlinear state-space form of the orbiting satellite dynamics and measurement is derived to be:

$$\mathcal{F}[x, u, \widetilde{w}] = \dot{x} = \begin{bmatrix} x_2 \\ -\dfrac{\mu x_1}{(x_1^2 + x_3^2)^{\frac{3}{2}}} + u_1 + \widetilde{w}_1 \\ x_4 \\ -\dfrac{\mu x_3}{(x_1^2 + x_3^2)^{\frac{3}{2}}} + u_2 + \widetilde{w}_2 \end{bmatrix}$$

$$h[x, u, \tilde{v}] = y = \begin{bmatrix} \sqrt{(x_1 - x_s^i)^2 + (x_3 - y_s^i)^2} \\ \dfrac{(x_1 - x_s^i)(x_2 - \dot{x}_s^i) + (x_3 - y_s^i)(x_4 - \dot{y}_s^i)}{\sqrt{(x_1 - x_s^i)^2 + (x_3 - y_s^i)^2}} \\ \tan^{-1}\left(\dfrac{x_3 - y_s^i}{x_1 - x_s^i}\right) \end{bmatrix}$$

The CT Jacobians can be calculated by taking the corresponding partial derivatives:

$$\tilde{A} = \left[\frac{\partial \mathcal{F}}{\partial x}\right]_{|nom} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\mu(2x_1^2 - x_3^2)}{(x_1^2 + x_3^3)^{\frac{5}{2}}} & 0 & \frac{3\mu(x_1 x_3)}{(x_1^2 + x_3^3)^{\frac{5}{2}}} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{3\mu(x_1 x_3)}{(x_1^2 + x_3^3)^{\frac{5}{2}}} & 0 & \frac{\mu(2x_3^2 - x_1^2)}{(x_1^2 + x_3^3)^{\frac{5}{2}}} & 0 \end{bmatrix}$$

$$\tilde{B} = \left[\frac{\partial \mathcal{F}}{\partial u}\right]|_{nom} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\Gamma = \left[\frac{\partial \mathcal{F}}{\partial \widetilde{w}}\right]|_{nom} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\tilde{C} = \left[\frac{\partial h}{\partial u}\right]|_{nom} =$$

$$\begin{bmatrix} \frac{x_1 - x_s}{[(x_1-x_s)^2+(x_3-y_s)^2]^{1/2}} & 0 & \frac{x_3 - y_s}{[(x_1-x_s)^2+(x_3-y_s)^2]^{1/2}} & 0 \\ \frac{(x_3-y_s)[(x_2-\dot{x}_s)(x_3-y_s)-(x_1-x_s)(x_4-\dot{y}_s)]}{[(x_1-x_s)^2+(x_3-y_s)^2]^{3/2}} & \frac{x_1-x_s}{[(x_1-x_s)^2+(x_3-y_s)^2]^{1/2}} & \frac{(x_1-x_s)[(x_1-x_s)(x_4-\dot{y}_s)-(x_2-\dot{x}_s)(x_3-y_s)]}{[(x_1-x_s)^2+(x_3-y_s)^2]^{3/2}} & \frac{x_3-y_s}{[(x_1-x_s)^2+(x_3-y_s)^2]^{1/2}} \\ \frac{y_s-x_3}{(x_1-x_s)^2+(x_3-y_s)^2} & 0 & \frac{x_1-x_s}{(x_1-x_s)^2+(x_3-y_s)^2} & 0 \end{bmatrix}$$

B. Discrete-Time Jacobians

The DT Jacobians can be calculated from the CT Jacobians for this system is shown below. The discrete time step for this orbit determination is taken to be $\Delta T = 10$ seconds. It can be seen that that these Jacobian matrices are time dependent:

$$\tilde{F} = I_{nxn} + \Delta T \tilde{A}$$
$$\tilde{G} = \Delta T \tilde{B}$$
$$\tilde{\Omega} = \Delta T \Gamma$$
$$\tilde{H} = \tilde{C}$$

C. Comparison Between Full Nonlinear Simulation and Linearized DT Dynamics Model

To get a better understanding of the system dynamics and how accurate a linearization of the system is, both the full, nonlinear system and linearized system is evaluated and compared against each other. The full nonlinear model was solved with Matlab's ODE45 function using the state space equations given by $\mathcal{F}[x, u]$. The linearized model was evaluated using the DT Jacobians to propagate an initial perturbation state. The perturbation states must be added to a known, nominal trajectory in order to recover the full linearized states. It is assumed that there is no control inputs and process noise for this section ($\boldsymbol{u}(t) = 0, \widetilde{\boldsymbol{w}}(t) = 0$).

Both the full nonlinear and linearized model have an initial perturbation of $\delta x = [0, 0.075, 0, -0.021]^T$ and initial state conditions $x_{init} = \left[r_0, 0, 0, r_0\sqrt{\frac{\mu}{r_0^3}}\right]^T$. The nominal state trajectory used for the linearized model has the nominal equations of motions given by:

$$\dot{x}_{nom} = \begin{bmatrix} x_2 \\ -\frac{\mu}{r_0^3} x_1 \\ x_4 \\ -\frac{\mu}{r_0^3} x_3 \end{bmatrix} \quad \Rightarrow \quad x_{tot,lin} = x_{nom} + \delta x$$

where the perturbation is calculated through the DT state transition matrix:

$$\delta x_{k+1} = \tilde{F}_k \delta x_k$$

Figure 1 below shows the comparison of the state propagation between the nonlinear system solved by ODE45 and the linearized model. Both systems were propagated to 14000 seconds, which is slightly over two orbital periods. It is difficult to see, but the two models begin with very close trajectories. However, as the systems evolve, there is a divergence that can be seen between all four states.

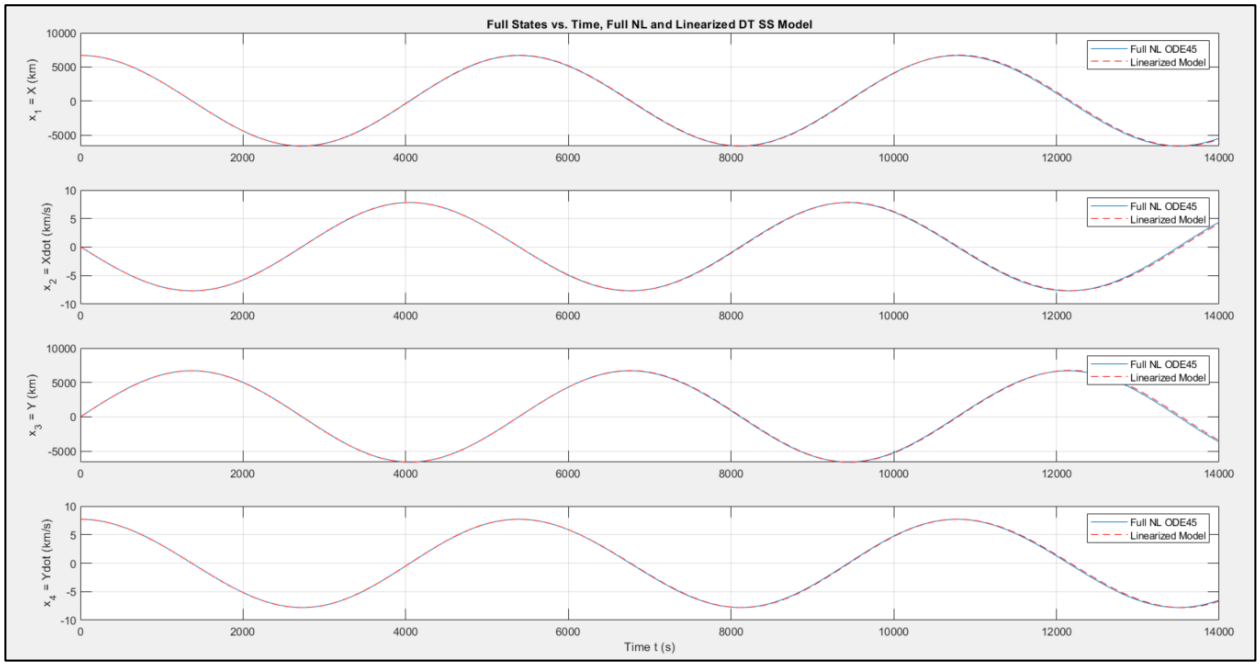

Fig. 1 State Comparison Between Nonlinear (ODE45) and Linearized Models

Figure 2 shows the errors between the nonlinear and linearized models. It is much more apparent in these plots that the two systems are increasingly divergent as time increases. In just slightly over two orbital periods, the position states have position errors as high as 200km, and the velocity states have velocity errors as up to 2km/s. However, these comparisons do show that the linearized model predicts the system dynamics relatively well within one orbital period.

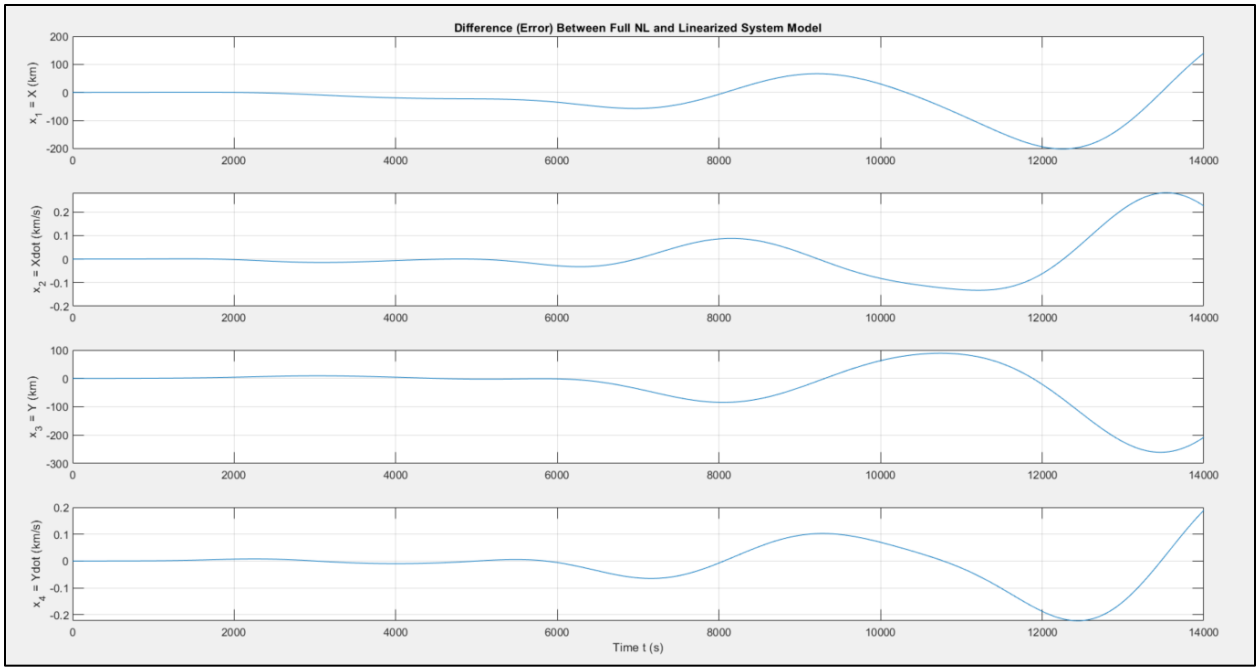

Fig. 2 State Errors Between Nonlinear (ODE45) and Linearized Models

D. Comparison Between Nonlinear and Linearized Ground Station Measurement Models

Similar to the nonlinear and linearized state dynamics model, the measurement model can also be linearized. The nonlinear measurements are determined by evaluating $h[x,u]$ with no measurement noise. The linearized measurement model calculates the perturbation measurement through the linearized perturbation state δx and $\tilde{H}$ matrix. The perturbation measurement is then added to the nominal measurement to recover the full measurement.

$$y_{tot,lin} = y_{nom} + \tilde{H}\delta x$$

Figure 3 and Figure 4 shows the nonlinear and linear measurement data with the associated tracking station number, respectively. Figure 5 shows the comparison between the nonlinear and linearized measurements for 14000 seconds. Like the linearized state model, it can be seen that the linearized measurement model eventually diverges more as time progresses. Figure 6 shows the errors between the two models. Since there are 12 ground tracking stations, the increase in errors could potentially cause issues in dissimilar stations providing measurement data.

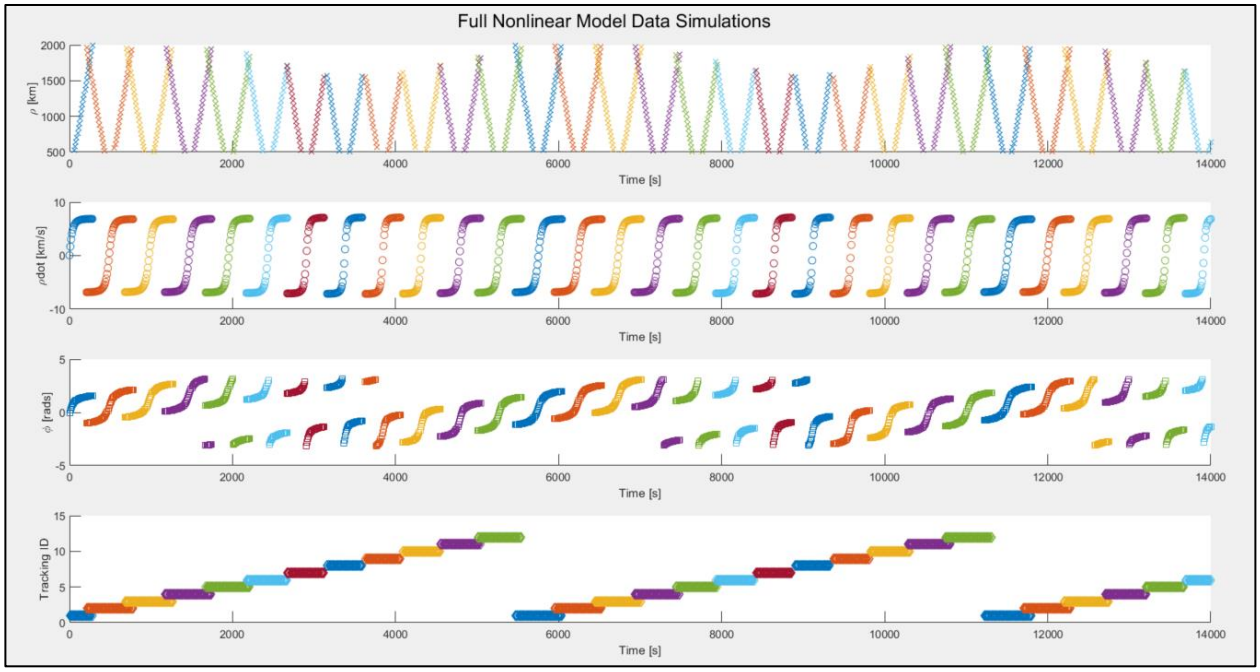

Fig. 3 Full Nonlinear Model Data Simulation

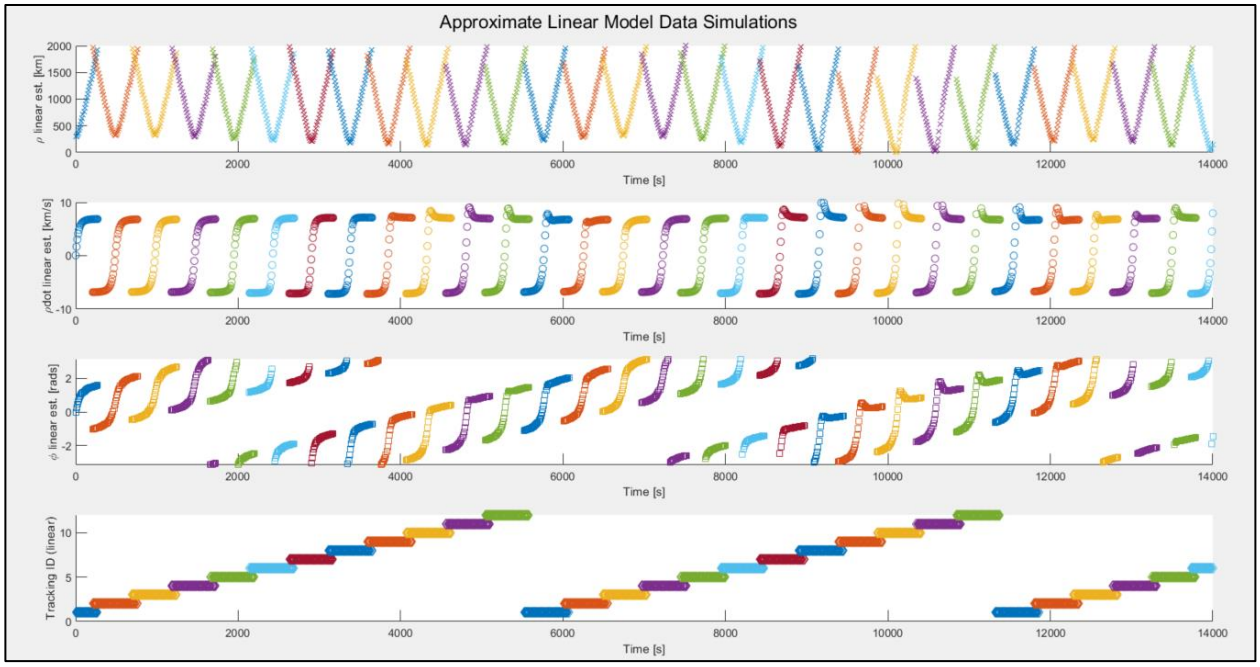

Fig. 4 Linearized Model Data Simulation

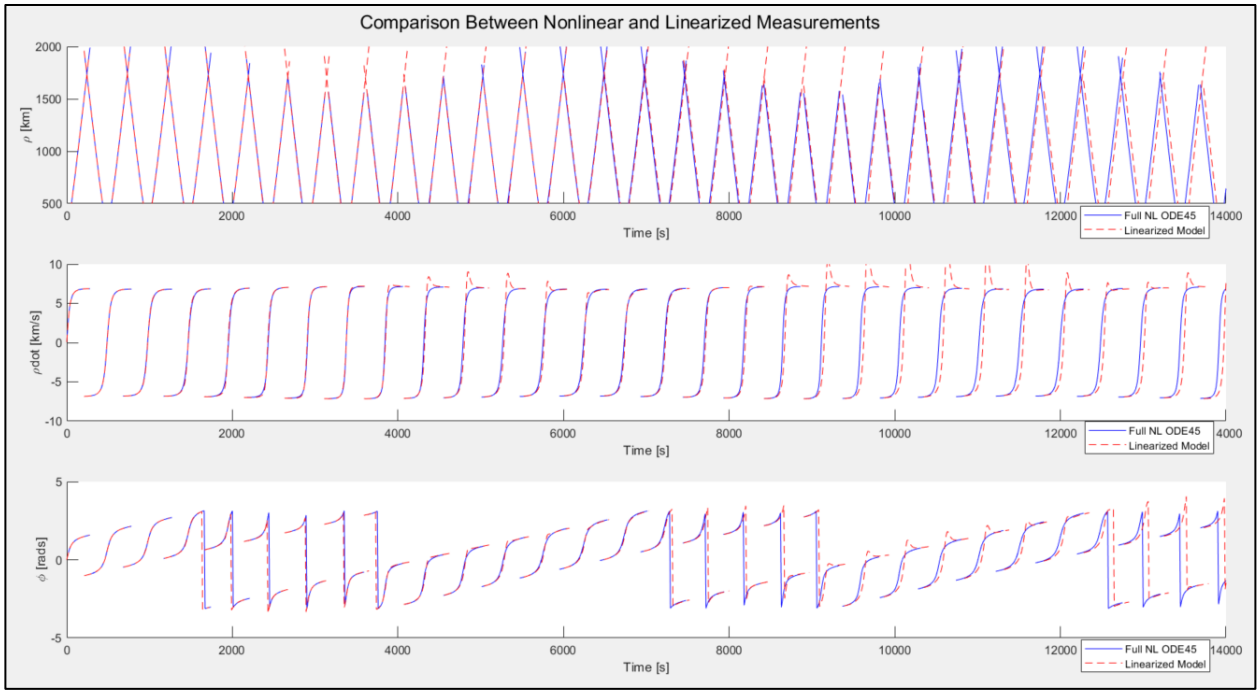

Fig. 5 Measurement Comparison Between Nonlinear and Linearized Models

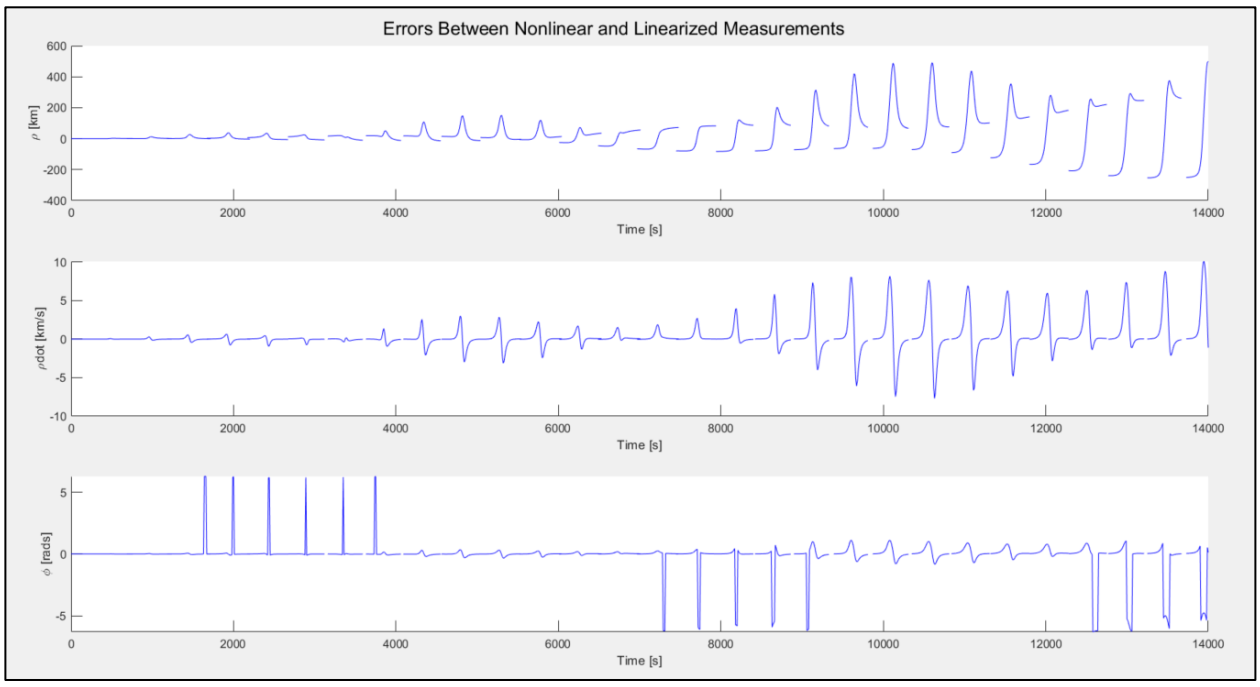

Fig. 6 Measurement Comparison Between Nonlinear and Linearized Models

IV. Section 2: Stochastic Nonlinear Filtering – Linearized Kalman Filter

The linearized Kalman filter centers around linearizing the dynamics and measurements of the system about a nominal trajectory. One benefit of the LKF is that the nominal trajectory can be calculated offline a-priori. One drawback, however, is the LKF will not work well if the true trajectory deviates too far from the nominal trajectory or if there are large process noises. The LKF utilizes the Eulerized Jacobians to estimate the state perturbation, δx. To recover the full state estimate, the state perturbation must be added back to the nominal trajectory (deterministic).

E. Typical Ground Truth Simulation for Noisy States and Measurements – LKF

In order to eventually validate the accuracy and robustness of the LKF, Monte Carlo truth model test (TMT) simulations need to be carried out to perform the NEES and NIS chi-square tests. Each Monte Carlo simulation requires a simulated noisy truth trajectory and corresponding noisy truth measurement model. The noisy trajectories and measurements are simulated by injecting process noise and measurement noise to the nominal trajectory with an initial perturbation of $\delta x = [0, 0.075, 0, -0.021]^T$.

The process noise is introduced to the nominal trajectory accelerations as additive white Gaussian noise (AWGN) through a Zero-Order Hold (ZOH) input. The noisy measurements are then determined from the noisy trajectory through the nonlinear equations given by $\boldsymbol{y}^i(t)$ with the measurement noise (AWGN) added. Both the process noise and measurement noises are randomly generated and sampled from the provided Q_{true} and R_{true} covariance matrices. The Cholesky decomposition and Matlab function *randn()* was used to add noise to the trajectory and measurements as shown:

$$S_w = chol(Q_{true}, 'lower')$$

$$S_v = chol(R_{true}, 'lower')$$

Figure 7 and Figure 8 shows a single, typical simulation instance of the noisy trajectory and noisy measurements, respectively. There are obvious differences in both the noisy trajectory and measurements compared to their respective nominals, including shifting.

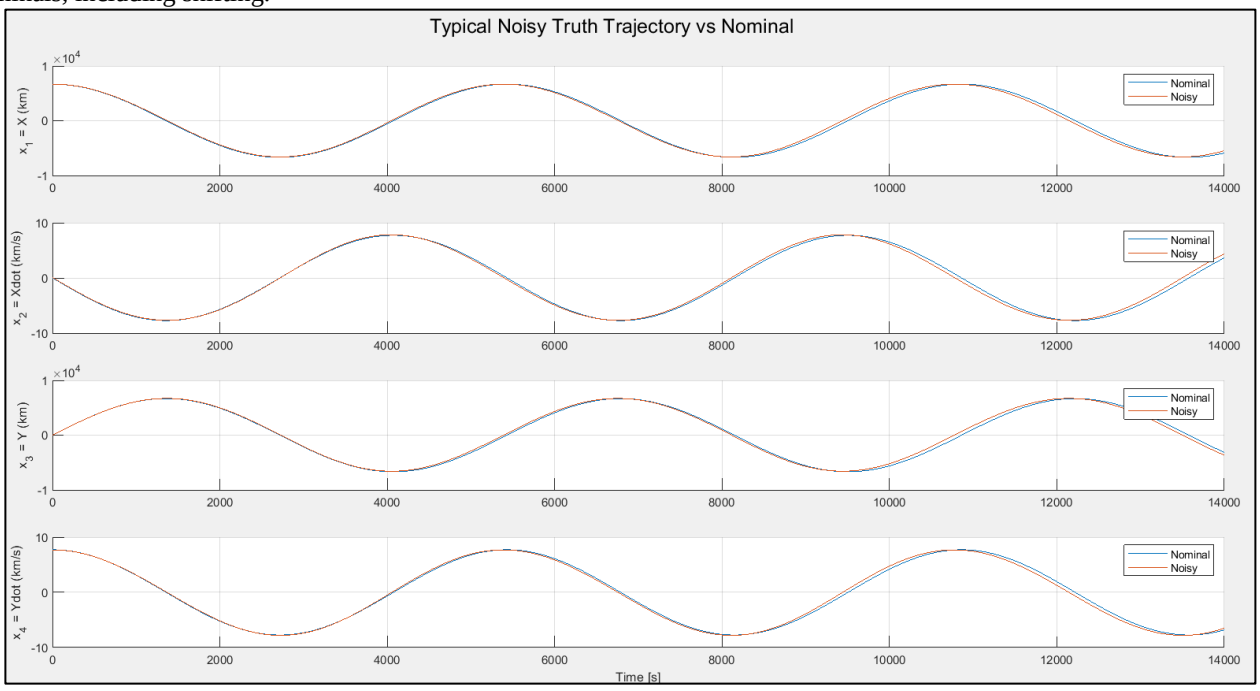

Fig. 7 Typical Noisy Trajectory vs Nominal Noiseless Trajectory

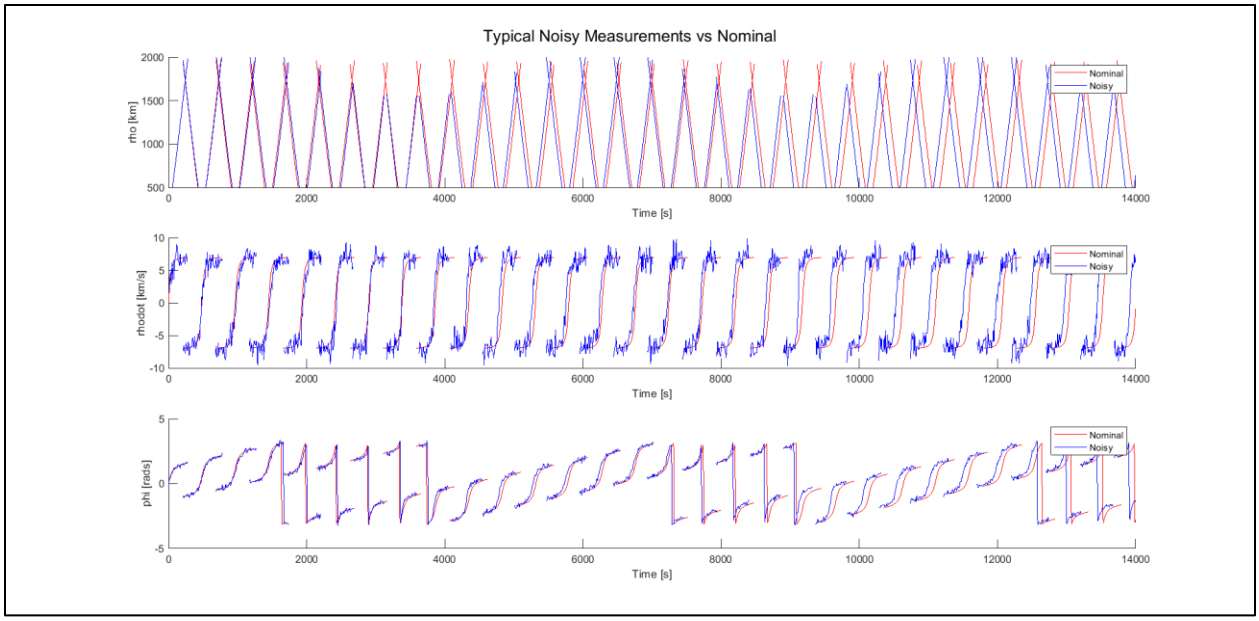

Fig. 8 Typical Noisy Measurements vs Nominal Noiseless Measurements

Figure 9 shows the typical LKF state estimation error for each state with their corresponding 2σ error bounds.. Figure 10 shows the same information, but for one orbital period in order to better see the responses. The state estimation errors ($e_{x,k}$) are calculated with respect to the ground truths (x_k) and the state estimations from the EKF ($\hat{x}_k^+$):

$$e_{x,k} = x_k - \hat{x}_k^+$$

The 2σ error bounds are calculated from the respective covariance estimates from the LKF:

$$\pm 2\sigma = 2\sqrt{diag(\hat{P}_k^+)}$$

While the LKF begins with low estimation error, as the nonlinearities of the system compound over time, the estimation error begins to blow up. Because the LKF linearizes about an offline nominal trajectory, it becomes a very poor estimator once the real trajectory has departed from the nominal. Because it is only checking the offline nominal, the LKF is not aware of the significant error and therefore the $\pm 2\sigma$ do not accurately represent the nature of the inaccuracy.

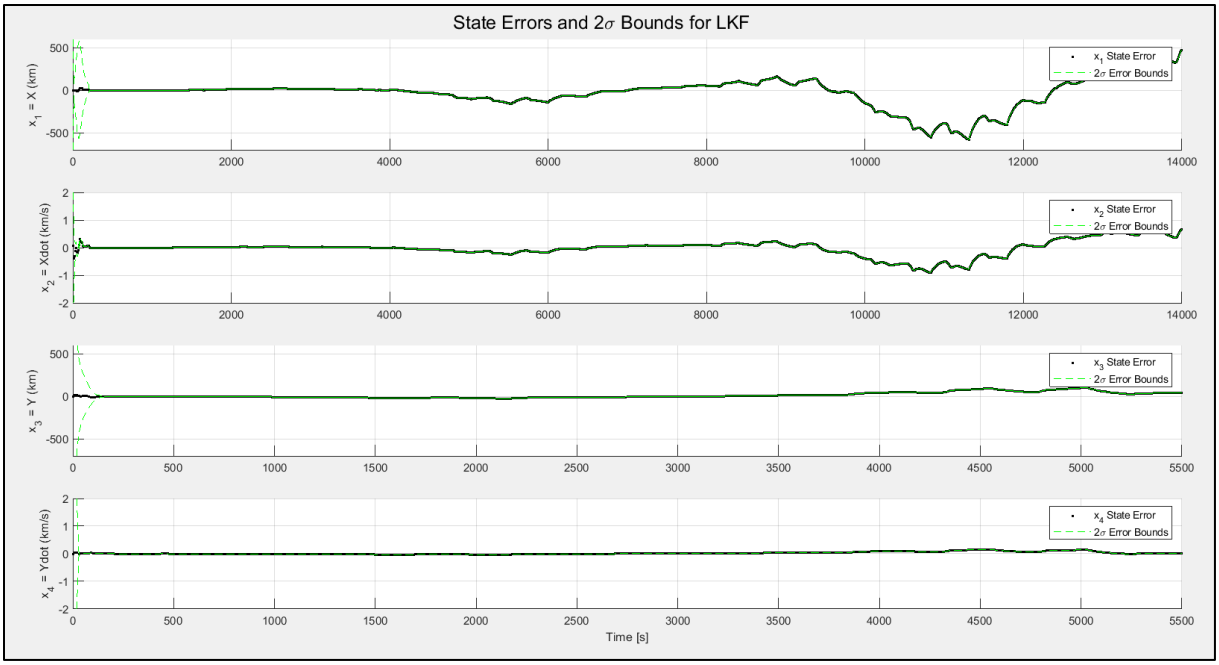

Fig. 9 Error Estimate from Linearized Kalman Filter for One Orbital Period

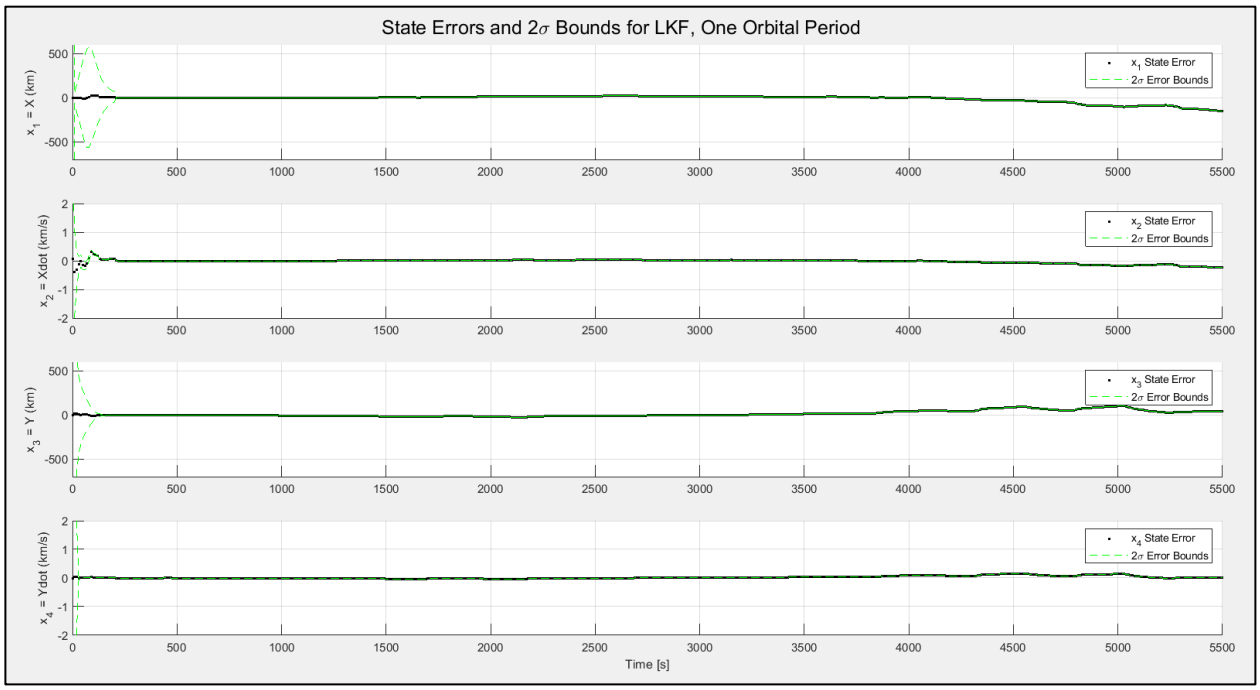

Fig. 10 Error Estimate from Linearized Kalman Filter for 14000 Seconds

F. NEES and NIS Statistics for Linear Kalman Filter

The NEES and NIS chi-squared tests are utilized to determine if the implemented linear Kalman filter's state errors and measurement residuals make sense for the system, measurement, and noise models. In other words, these tests check if the actual state errors and measurement residuals agrees with the linear Kalman filter's estimated error covariance. It can be shown that if the LKF is working properly based on the DT state space model and noise parameters, then there are two conditions that must be met:

I. NEES Condition: if $\epsilon_{x,k} = e_{x,k}^T (P_k^+)^{-1} e_{x,k}$, where $e_{x,k}^T = x_k - \hat{x}_k^+$,

$$\rightarrow \text{then } \epsilon_{x,k} \sim \chi_n^2 \; \forall \; k, \text{ where } E[\epsilon_{x,k}] = n, \; var(\epsilon_{x,k}) = 2n$$

II. NIS Condition: if $\epsilon_{y,k} = e_{y,k}^T (S_k)^{-1} e_{y,k}$, where $e_{y,k} = y_k - \hat{y}_k^-$,

$$\rightarrow \text{then } \epsilon_{y,k} \sim \chi_p^2 \; \forall \; k, \text{ where } E[\epsilon_{y,k}] = p, \; var(\epsilon_{y,k}) = 2p$$

Each NEES and NIS data point at time k is the resultant average of N Monte Carlo simulation runs of each of the randomly generated noisy truth trajectories, which were run through the LKF. In order to keep the runs consistent, the same initial state and covariance guesses were used for all Monte Carlo simulations. The initial state was chosen to be:

$$\hat{x}^+(0) = \left[r_0, 0, 0, r_0 \sqrt{\frac{\mu}{r_0^3}} \right]^T$$

The initial covariance matrix used the "inflated" diagonal approach for choosing the values to use. To keep the values consistent, the position variances (x_1, x_3) were chosen to be 20% r_0. The velocity variances (x_2, x_4) were chosen to be 20% $r_0 \sqrt{\frac{\mu}{r_0^3}}$. Thus, the initial covariance matrix used was:

$$\hat{P}^+(0) = 0.2 diag\left(\left[r_0, r_0 \sqrt{\frac{\mu}{r_0^3}}, r_0, r_0 \sqrt{\frac{\mu}{r_0^3}} \right] \right)$$

Other parameters of the NEES and NIS tests to consider are the number of Monte Carlo runs and the confidence levels. $N = 10$ Monte Carlo runs were chosen for simulation time considerations. There did not appear to be a significant statistical difference in results if $N > 10$ runs were chosen. $\alpha = 0.05$ was chosen, which gives this study a 95% confidence bounds.

In order to tune the LKF, the process noise Q_{KF} can be tuned. The approach for tuning Q_{KF} was to choose an upper and lower bound value and determine the appropriate values based on the NEES and NIS plots. While there were more iterations, only the lower bound and final Q_{KF} iterations are shown in this paper for conciseness.

1. $Q_{KF} = 10^{-5} I_{2x2}$ (Upper Guess)

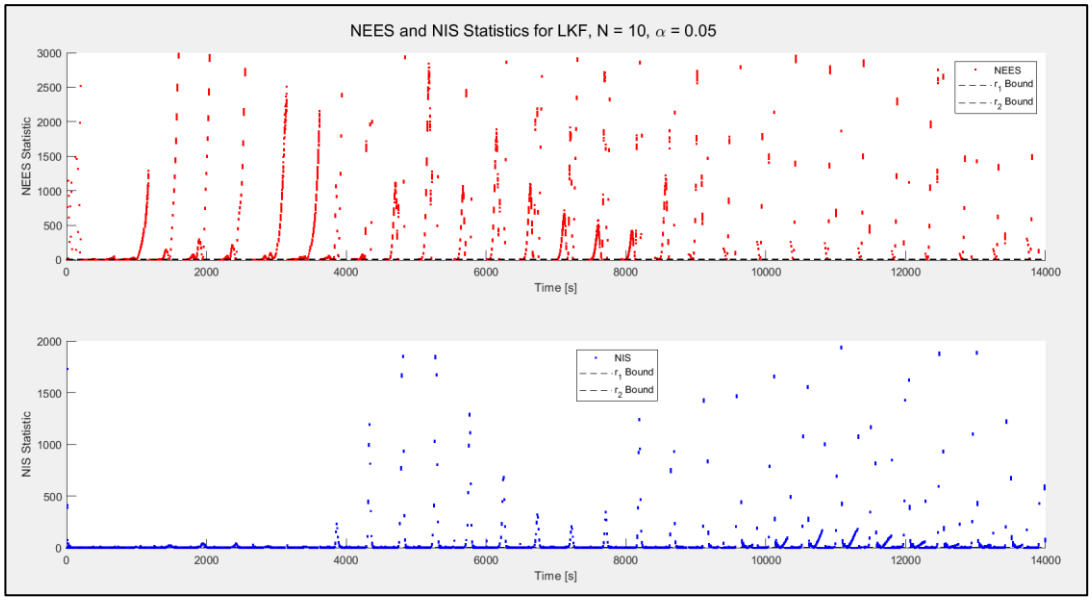

Fig. 10 NEES & NIS Statistics for $Q_{KF} = 10^{-5} I_{2x2}$ for LKF

Figure 11 shows that that a majority of the points fall very far above the upper confidence bound. This would suggest that the LKF is too optimistic. However, larger values for Q_{KF} only exacerbated this problem, so smaller values of Q_{KF} were tried (to little success).

2. $Q_{KF} = 10^{-10} I_{2x2}$ (Lower Guess)

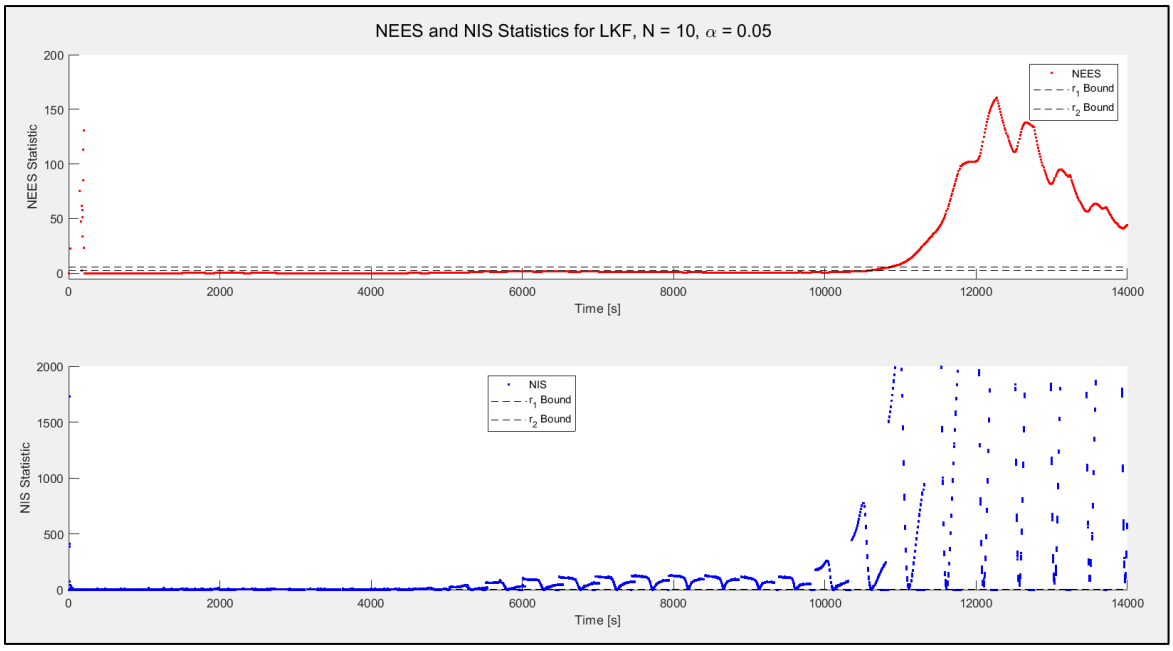

Fig. 12 NEES & NIS Statistics for $Q_{KF} = 10^{-10} I_{2x2}$ for LKF

3. $Q_{KF} = 10^{-10} I_{2x2}$ (Lower Guess)

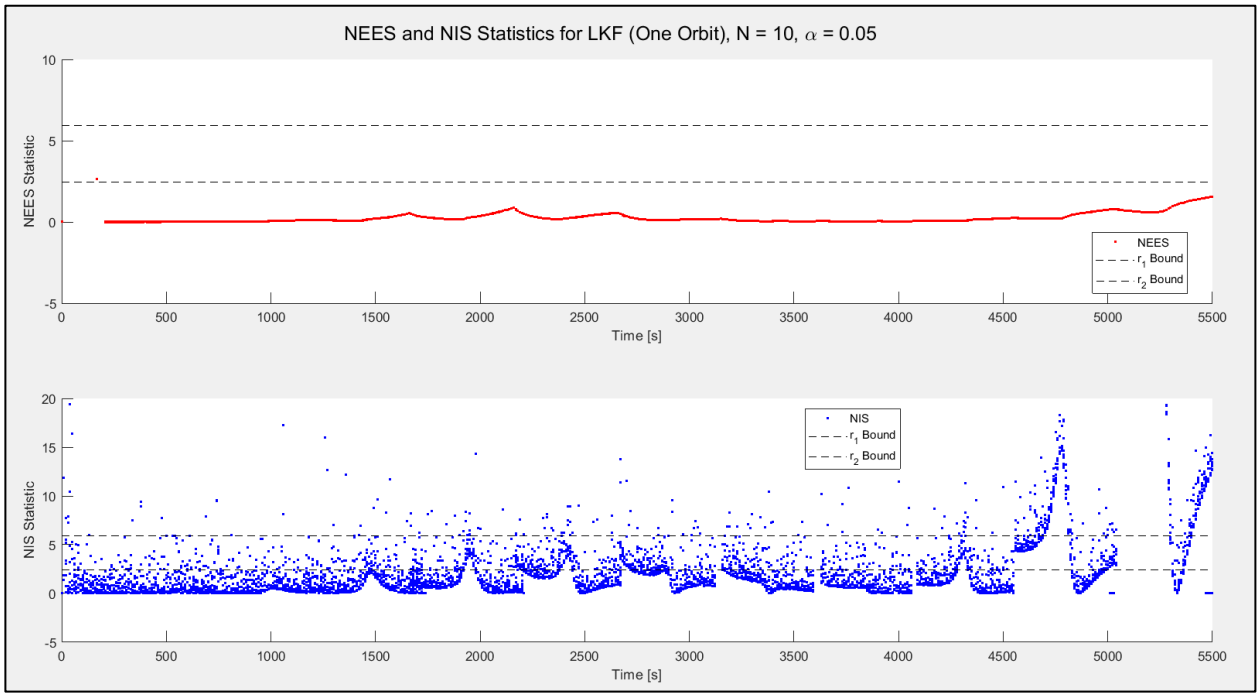

Fig. 13 NEES & NIS Statistics for $Q_{KF} = 10^{-10} I_{2x2}$ for LKF (One Orbital Period)

Unsurprisingly, it was not possible to tune the LKF to an acceptable validity level. Due to the unstable state estimation error, there appears to be no Q_{KF} value to which the LKF can be tuned to pass the truth model tests. The NEES and NIS statistics for $Q_{KF} = 10^{-10} I_{2x2}$ look better than for $Q_{KF} = 10^{-5} I_{2x2}$, however are far from passing.

V. Section 3: Stochastic Nonlinear Filtering – Extended Kalman Filter

The extended Kalman filter differs from the linearized Kalman filter in that it does not require a pre-calculated nominal trajectory. Instead, the EKF recalculates the nominal trajectory "online" using the full nonlinear dynamics at each time step and linearizes the state Jacobians based on the most up-to-date state estimate. Furthermore, the EKF estimates the full state, not the perturbation state like the LKF. This can potentially make the EKF more robust to small perturbations, as the nominal trajectory is optimized for each time step.

G. Typical Ground Truth Simulation for Noisy States and Measurements – EKF

In order to eventually validate the accuracy and robustness of the EKF, Monte Carlo truth model test (TMT) simulations need to be carried out to perform the NEES and NIS chi-square tests. Each Monte Carlo simulation requires a simulated noisy truth trajectory and corresponding noisy truth measurement model. The noisy trajectories and measurements are simulated by injecting process noise and measurement noise to the nominal trajectory with an initial perturbation of $\delta x = [0, 0.075, 0, -0.021]^T$.

The process noise is introduced to the nominal trajectory accelerations as additive white Gaussian noise (AWGN) through a Zero-Order Hold (ZOH) input. The noisy measurements are then determined from the noisy trajectory through the nonlinear equations given by $\boldsymbol{y}^i(t)$ with the measurement noise (AWGN) added. Both the process noise and measurement noises are randomly generated and sampled from the provided Q_{true} and R_{true} covariance matrices. The Cholesky decomposition and Matlab function *randn()* was used to add noise to the trajectory and measurements as shown:

$$S_w = chol(Q_{true}, 'lower')$$

$$S_v = chol(R_{true}, 'lower')$$

Figure 14 and Figure 15 shows a single, typical simulation instance of the noisy trajectory and noisy measurements, respectively. There are obvious differences in both the noisy trajectory and measurements compared to their respective nominals, including shifting.

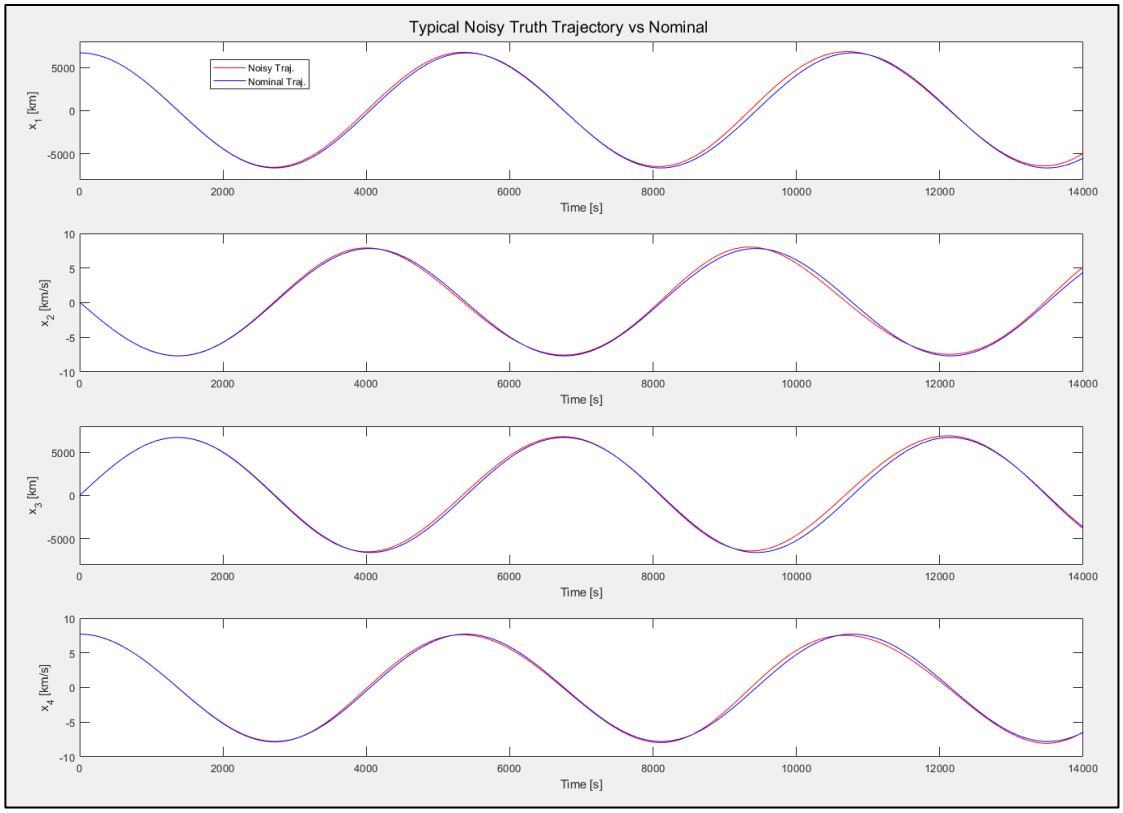

Fig. 14 Typical Noisy Trajectory vs Nominal Noiseless Trajectory

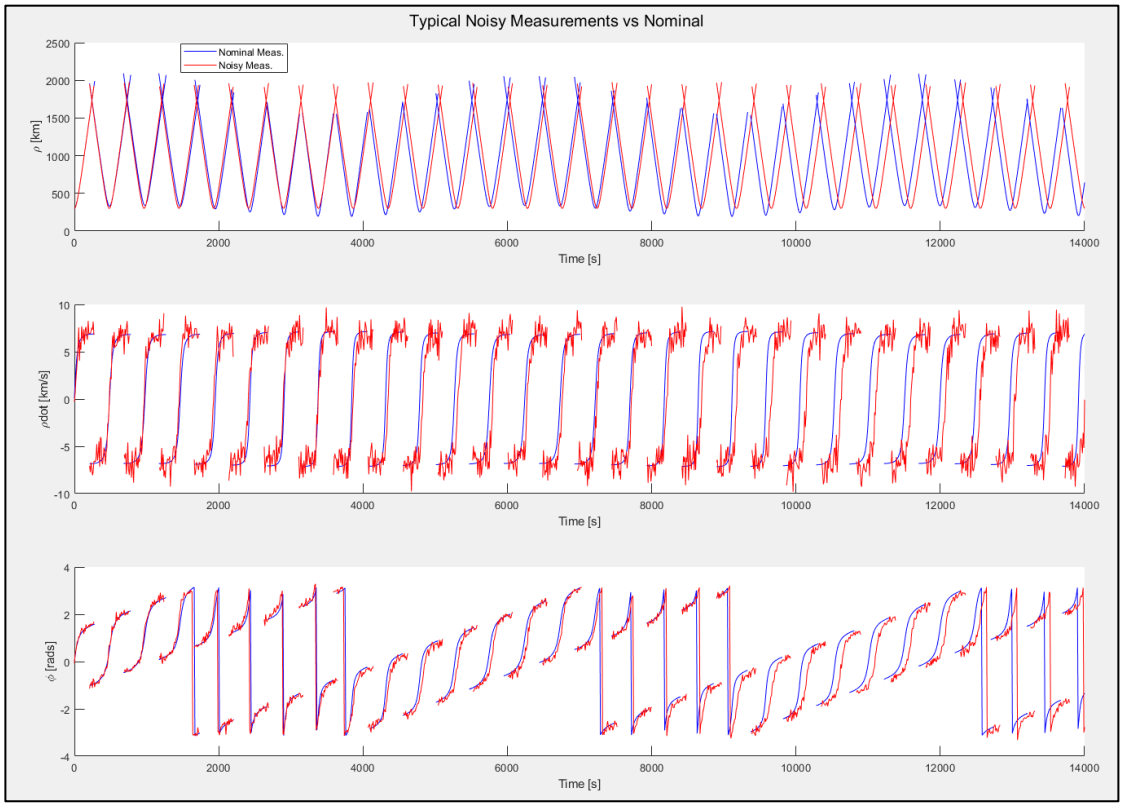

Fig. 15 Typical Noisy Measurements vs Nominal Noiseless Measurements

Figure 16 shows the typical EKF state estimation error for each state with their corresponding 2σ error bounds. Figure 17 shows the same information, but for one orbital period in order to better see the responses. The state estimation errors ($e_{x,k}$) are calculated with respect to the ground truths (x_k) and the state estimations from the EKF ($\hat{x}_k^+$):

$$e_{x,k} = x_k - \hat{x}_k^+$$

The 2σ error bounds are calculated from the respective covariance estimates from the EKF:

$$\pm 2\sigma = 2\sqrt{diag(\hat{P}_k^+)}$$

For all four state errors, the 2σ error bounds begin wider, but quickly converges down. Compared to the LKF, the EKF appears to be more robust and has a better capability of estimating the trajectory which includes a perturbation from nominal. This can be attributed to the EKF constantly updating the nominal trajectory at each time step. It does not depend on a pre-calculated nominal trajectory, which can cause issues if the true trajectory strays too far away from nominal. It can be seen that the state estimation errors for the EKF stay within the 2σ error bounds well.

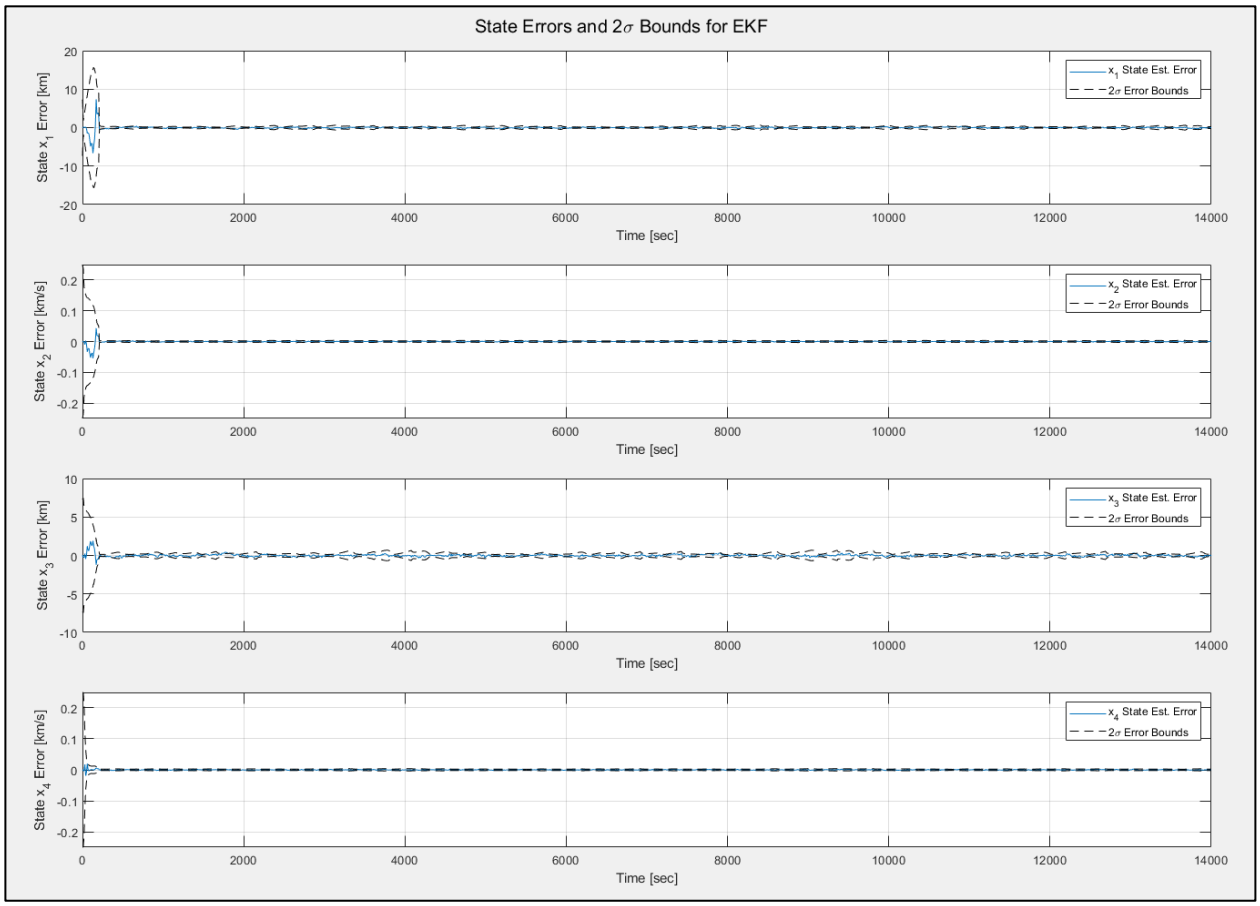

Fig. 16 Error Estimate from Extended Kalman Filter for 14000 Seconds

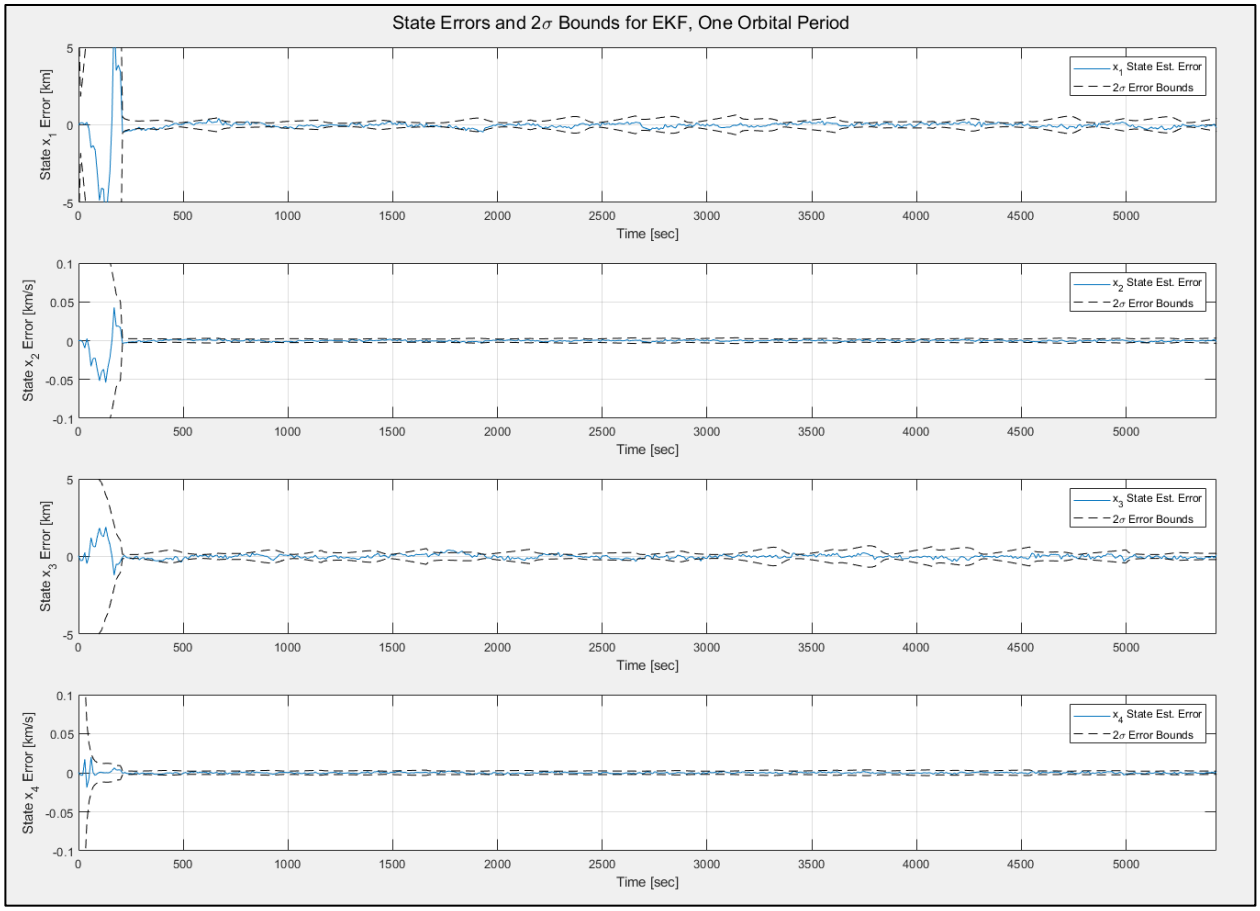

Fig. 17 Error Estimate from Extended Kalman Filter for One Orbital Period

H. NEES and NIS Statistics for Extended Kalman Filter

The NEES and NIS chi-squared tests are utilized to determine if the implemented extended Kalman filter's state errors and measurement residuals make sense for the system, measurement, and noise models. In other words, these tests check if the actual state errors and measurement residuals agrees with the extended Kalman filter's estimated error covariance. It can be shown that if the EKF is working properly based on the DT state space model and noise parameters, then there are two conditions that must be met:

III. NEES Condition: if $\epsilon_{x,k} = e_{x,k}^T (P_k^+)^{-1} e_{x,k}$, where $e_{x,k}^T = x_k - \hat{x}_k^+$,

$$\rightarrow \text{then } \epsilon_{x,k} \sim \chi_n^2 \; \forall \; k, \text{ where } E[\epsilon_{x,k}] = n, \; var(\epsilon_{x,k}) = 2n$$

IV. NIS Condition: if $\epsilon_{y,k} = e_{y,k}^T (S_k)^{-1} e_{y,k}$, where $e_{y,k} = y_k - \hat{y}_k^-$,

$$\rightarrow \text{then } \epsilon_{y,k} \sim \chi_p^2 \; \forall \; k, \text{ where } E[\epsilon_{y,k}] = p, \; var(\epsilon_{y,k}) = 2p$$

Each NEES and NIS data point at time k is the resultant average of N Monte Carlo simulation runs of each of the randomly generated noisy truth trajectories, which were run through the EKF. In order to keep the runs consistent, the same initial state and covariance guesses were used for all Monte Carlo simulations. The initial state was chosen to be:

$$\hat{x}^+(0) = \left[r_0, 0, 0, r_0\sqrt{\frac{\mu}{r_0^3}}\right]^T$$

The initial covariance matrix used the "inflated" diagonal approach for choosing the values to use. To keep the values consistent, the position variances (x_1, x_3) were chosen to be 20% r_0. The velocity variances (x_2, x_4) were chosen to be 20% $r_0\sqrt{\frac{\mu}{r_0^3}}$. Thus, the initial covariance matrix used was:

$$\hat{P}^+(0) = 0.2diag\left(\left[r_0, r_0\sqrt{\frac{\mu}{r_0^3}}, r_0, r_0\sqrt{\frac{\mu}{r_0^3}}\right]\right)$$

Other parameters of the NEES and NIS tests to consider are the number of Monte Carlo runs and the confidence levels. For the EKF, $N = 10$ Monte Carlo runs were chosen for simulation time considerations. There did not appear to be a significant statistical difference in results if $N > 10$ runs were chosen. $\alpha = 0.05$ was chosen, which gives this study a 95% confidence bounds.

In order to tune the EKF, the process noise Q_{KF} can be tuned. The approach for tuning Q_{KF} was to choose an upper and lower bound value and determine the appropriate values based on the NEES and NIS plots. While there were more iterations, only the lower bound and final Q_{KF} iterations are shown in this paper for conciseness.

1. $Q_{KF} = 10^{-5}I_{2x2}$ (Upper Guess)

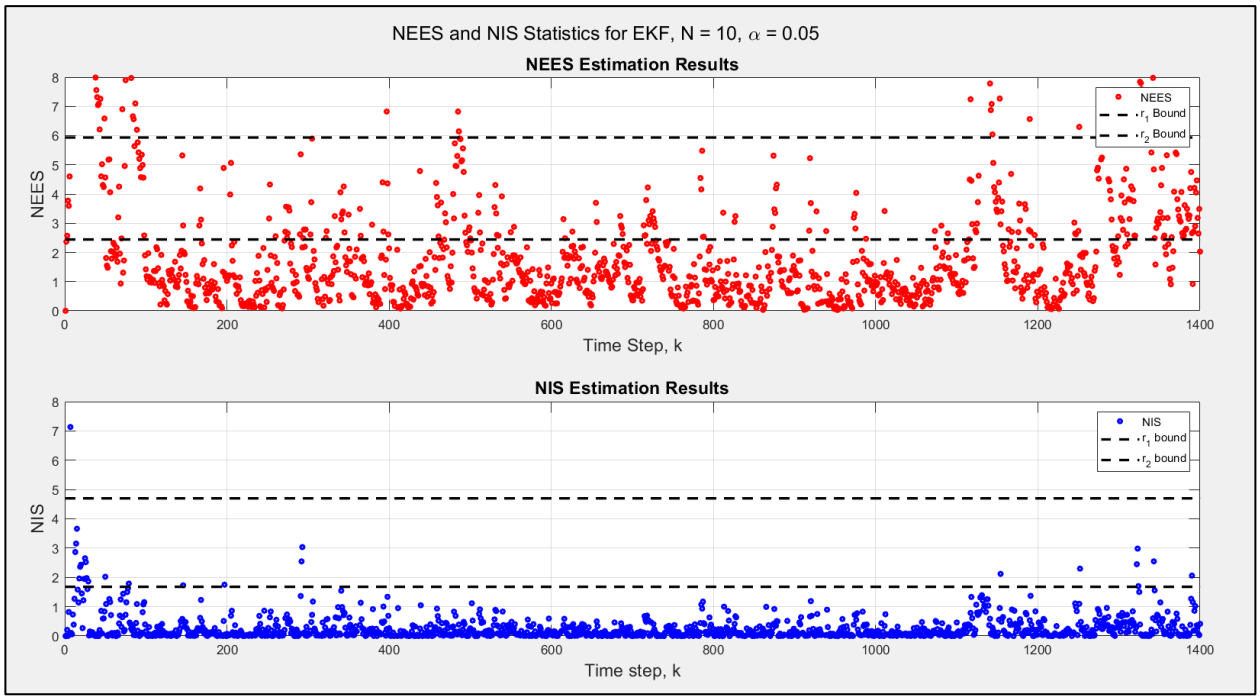

Fig. 18 NEES & NIS Statistics for $Q_{KF} = 10^{-5}I_{2x2}$ for EKF

From the NEES and NIS statistics for $Q_{KF} = 10^{-5}I_{2x2}$, it can be observed that a majority of the points fall well below the lower confidence bound. This suggests that the EKF is overly conservative and pessimistic. In this case, it can be concluded that Q_{KF} is too large. Thus, the value must be tuned smaller, as shown in the next section.

2. $Q_{KF} = 10^{-10} I_{2x2}$ (Final Guess)

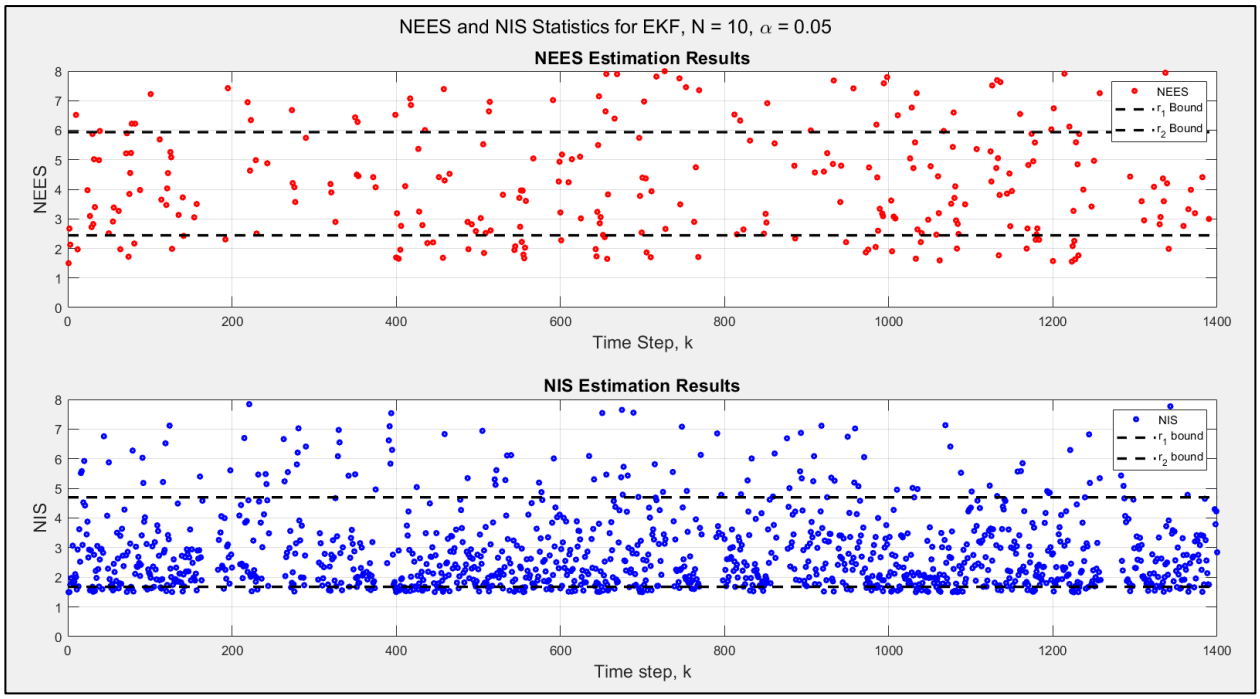

Fig. 19 NEES & NIS Statistics for $Q_{KF} = 10^{-10} I_{2x2}$ for EKF

The NEES and NIS statistics for $Q_{KF} = 10^{-10} I_{2x2}$ shows a much better distribution around the expected values ($n = 4$ for NEES, $p = 3$ for NIS) and variances ($2n$ for NEES, $2p$ for NIS). While there can be even more refined tuning for Q_{KF}, there can also be other contributing factors outside of tuning. For example, there can be significant, persistent biases that result from neglected higher order terms in the dynamics. However, based on these results, it can be concluded that the process noise $Q_{KF} = 10^{-10} I_{2x2}$ is a reasonable estimate for the implemented EKF.

VI.State Trajectory Estimation for LFK and EKF for Observation Data Log

The following section compares the state estimation from the implemented and tuned LKF and EKF developed from the previous sections. These filters are used to estimate the states from the measurements provided in the data log of the problem statement. Figure 20 and Figure 22 show the state estimates for 14000 seconds (slightly over 2 orbital periods) for the LKF and EKF, respectively. Figure 21 and Figure 23 show the state estimates for one orbital period for the LKF and EKF, respectively. All four plots also include the 2σ error bounds for each state.

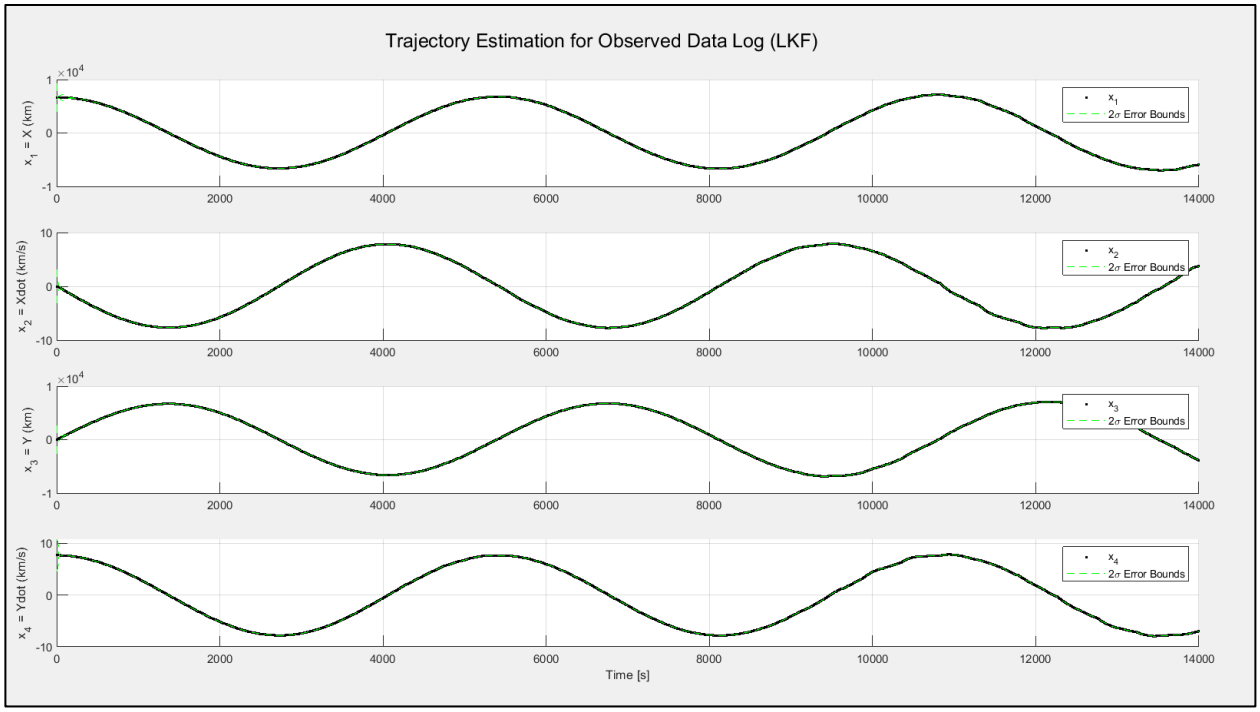

Fig. 20 LKF Estimated Trajectory of Given Data Log for 14000 seconds

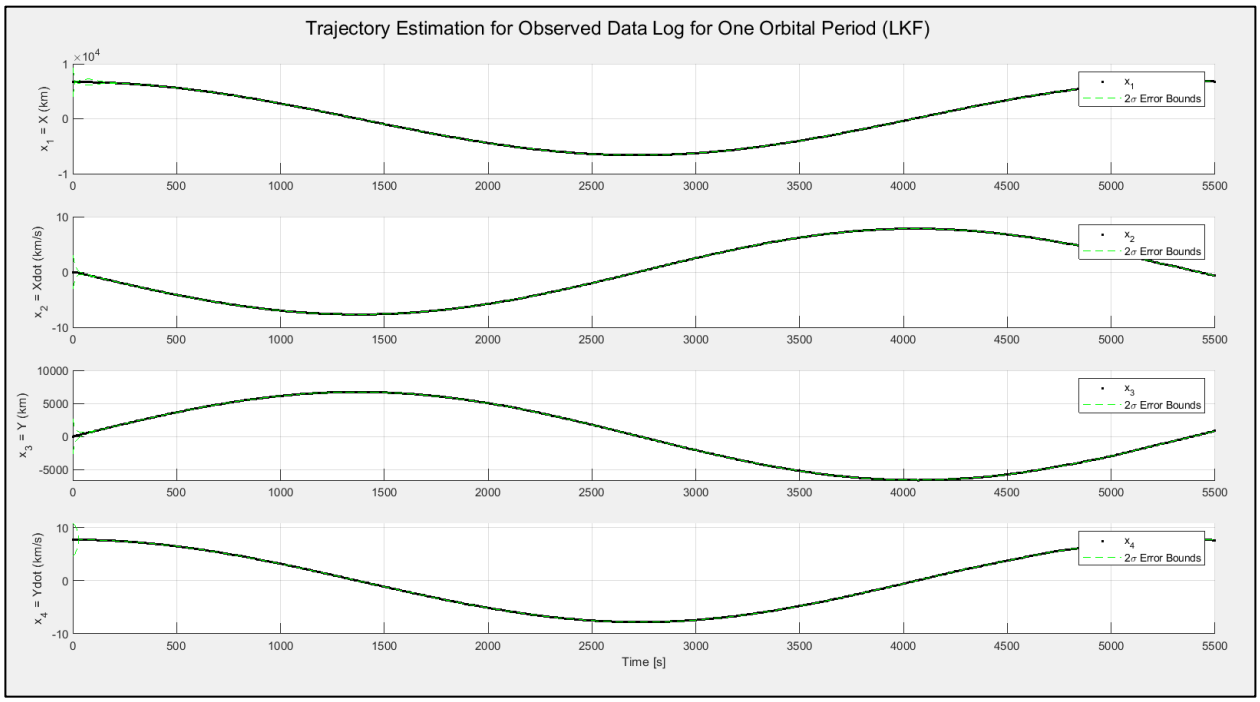

Fig. 21 LKF Estimated Trajectory of Given Data Log for One Orbital Period

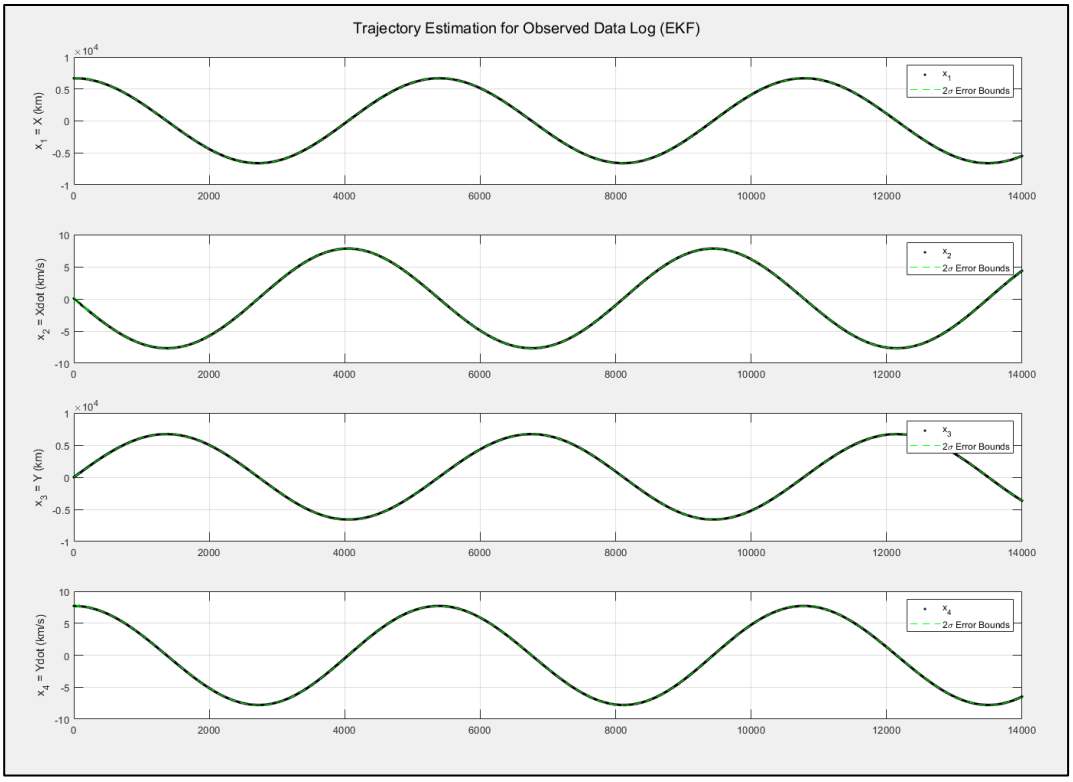

Fig. 22 EKF Estimated Trajectory of Given Data Log for 14000 seconds

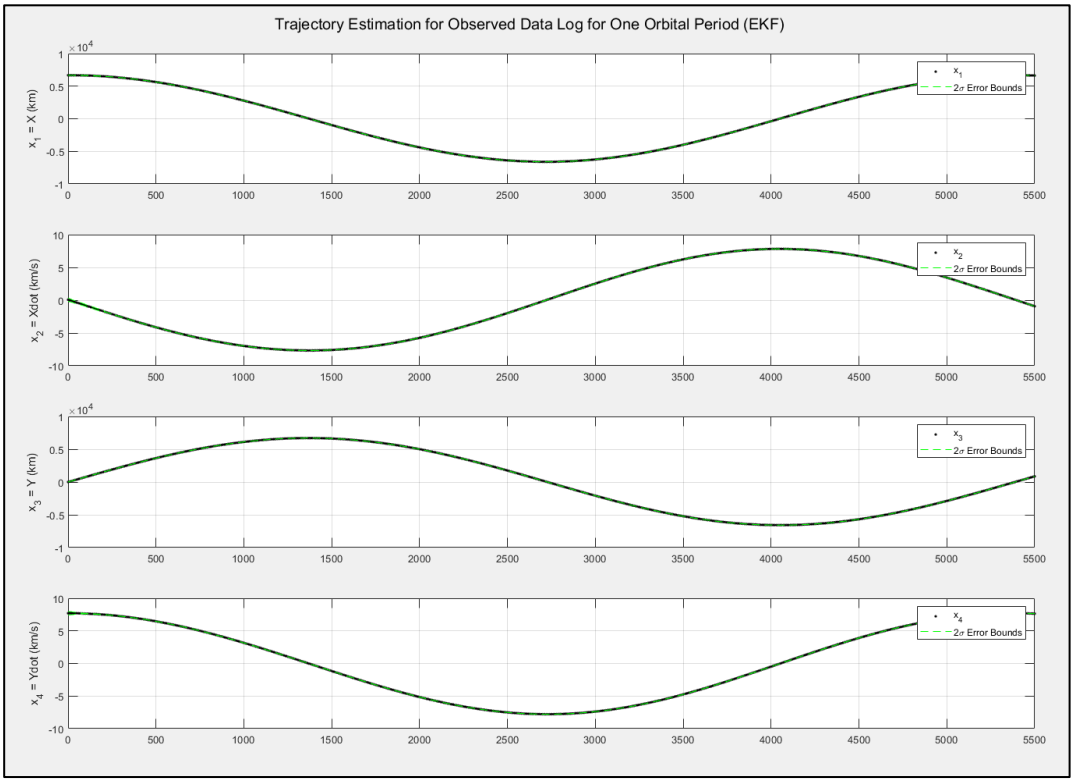

Fig. 23 EKF Estimated Trajectory of Given Data Log for One Orbital Period

Comparing Figure 20 and Figure 22, it is noticeable that there is significant noise appearing for the LKF near the beginning of the second orbital period (10000 seconds). The EKF, however, appears to continue tracking smoothly after that time period. Examining the results for just the first orbital period (Figure 21, Figure 23), both filters appear to estimate the trajectory equally well.

This difference in the LKF and EKF performance is likely attributed to the difference between the nominal and true trajectory (in the case of LKF). As the true trajectory progresses, the state perturbations increase, which leads to further deviation from the nominal trajectory. The EKF does not show this issue, as it is continuously recalculating the best estimate for the trajectory at each time step.

VII.Conclusion

The implementation of the predicator-corrector class of estimators known as Kalman filters were explored in this paper. Both the linearized Kalman filter and extended Kalman filter were created, tuned, and implemented to compare the performance and capability for a simplified statistical orbit determination problem.

Based on the results of implementing the LKF and EKF for the given measurements in the data log, it is concluded that the LKF performs well with small perturbations and being close to the nominal trajectory. In this specific case, the LKF performs well up to one orbital period. The EKF performs well for the full duration tested (over two orbital periods). The difference between the filter performances shows that the EKF is more robust to perturbations in the system and is overall a better filter to be used for this orbit determination problem.

VIII. AQ13 – An Intermission ~ Haiku

Unknown and fuzzy,

Your mystery intrigues me –

Kalman reveals you.

IX. AQ14 Unscented Kalman Filter / Sigma Point Filter

While the LKF and EKF can be powerful methods of estimating nonlinear systems, there are still many limitations and drawbacks that arise in real systems. Some instances where the LKF and EKF may not be ideal for use is when the true trajectory of the system diverges from the estimate (LKF), if there are higher order dynamics that become significant, or if the dynamics are non-differentiable. The unscented Kalman filter (UKF) can be a good alternative for these instances. The key idea in the UKF is to directly approximate the mean and covariance of the system from one pdf to the next instead of analytically approximating the nonlinear state transition matrix first.

The UKF utilizes specific sigma points $(2n + 1)$ beginning from the initial state pdf for state x_k, and propagates those sigma points through the nonlinear dynamics to simulate the next estimated state x_{k+1}. One of the largest benefits of the UKF over the LKF and EKF is that it does not require the calculation of Jacobians. The UKF is able to retain higher order terms, which is essential if those become significant in the dynamics of the system.

I. UKF Results and Discussion Section

An attempt was made to implement the unscented Kalman filter. However, after a single timestep, the resulting covariances become non positive-definite; therefore, its Cholesky decomposition cannot be found. This could be remedied by implementing a square-root unscented Kalman filter.

X.References

[1] Simon, D., *Optimal state estimation: Kalman, H and nonlinear approaches*, Hoboken: Wiley-Interscience, 2006.

[2] Ahmed, N., "Lectures 01-33," *University of Boulder - Colorado, ASEN 5044, Fall 2020*, 2020.