# Question #5

## Extended Kalman Filter (EKF)

The EKF starts off at time-step zero (0) with an estimate of the total state and covariance. At each time-step k, during the prediction step the EKF uses the non-linear equations to calculate the estimated state which in turn is used to approximate the predicted covariance matrix.

,

where and

During the correction step the EKF calculates the innovation vector () using the measurement estimate derived from the non-linear measurement equation.

The filter then generates a new DT System (H) matrix, linearized about the new state estimate, to calculate the Kalman gain, update the total state estimate and update the covariance matrix.

,

where

MATLAB code for implementation and tuning of the EKF can be found in Appendix E.

## Tuning Approach

Tuning the filter involved generating randomized ground truth data for Monte Carlo simulations and calculating the mean NEES and NIS scores at each time step. The EKF’s process noise matrix Q was then tweaked until the NEES and NIS consistency tests were within the desired accuracy bounds (alpha = 0.01). The list below captures guidelines for and lessons learned during tuning.

***Tuning Guidelines / Lessons Learned***

* Use State Error plots to tune individual Q diagonal gains. If more than 3% of signal exceeds 2-sigma bounds, increase Q gain (filter is over-confident on state). If less than 3% the filter is too pessimistic, decrease Q.
* If periodicity of error signal is slow need to increase time duration of test or increase number of Monte Carlo runs, else testing might not be capturing long-term behavior of filter. The time duration was set to 100 seconds so multiple UAV orbits were included in a run and because the UGV error signal has low-frequency content (likely related to UAV orbit).
* The initial position should be randomized using the initial covariance matrix to prevent odd NEES/NIS behavior at the start. Use the error bounds to estimate the average covariance and use this as an initial guess.
* If the NEES scores are above the upper bound the Q values are too small (over-confident) and if the Q values are too small the NEES scores will be below the lower bound.
* The alpha value—desired filter consistency value—is dependent on the use cause but 5-10% is generally acceptable.
* The chi-square scores seemed to be most sensitive to the UGV and UAV angle noise and least sensitive to the UAV north/east position noise.
* Since the vehicle position/velocity are a function of the heading angle, there’s some coupling between these states so their off-diagonals were made non-zero.
* Creating a plot with state error plots for all runs was informative for making tuning decisions.

## Typical Simulation Plots

The section shows plots from a typical simulation run. Figure 1 captures an example of the randomly generated UGV and UAV state data.



Figure - Simulation Run - Ground Truth States

Figure 2 shows an example of the randomly generated noisy measurement data.



Figure - Simulation Run - Simulated Measurement Data

Figure 3 is and example of the state estimate errors; the ±2σ bounds calculated using the covariance matrix are shown as red dotted lines.



Figure – Estimated State Errors

An example of the typical measurement error is shown in Figure 4.



Figure - Estimated Measurement Errors

## Chi-Square Test Results

This section shows the final NEES and NIS test results for 100 Monte Carlo simulation runs and an alpha value of 5%. Our filter passes the NEES test criteria as the mean score is approximately equal to the number of states (n = 6) and approximately 5% of samples are beyond the limits.



Figure - Final NEES results



Figure - Final NIS Results