Problem Set 7

AA 279D, Spring 2018

Due: May 30, 2018 (Wednesday) at 1:30 pm

Notes:

As with the previous problem set, this assignment will take a more open-ended approach than previous ones. You will be guided through the design and implementation of a representative relative navigation system for your mission.

Submission Instructions

Each student should complete Part 1 of this problem set by the due date. You will not be expected to complete Part 2 within the week, however you should demonstrate progress on the filter design and implementation if you wish to receive feedback on your direction.

Part 1. Navigation System Design

As discussed in class, the prevailing mechanism for aerospace navigation systems is the extended Kalman filter (EKF). As such, you will design and implement an EKF to estimate the relative state of your formation in this assignment. There are several important parameters to consider along the way:

- What relative state representation will you estimate with your filter? Examples we have seen in class include the Cartesian relative state, orbit element differences, and quasi-nonsingular relative orbit elements.
- What dynamics model will you use to perform the EKF state prediction? Will you perform the prediction with a closed-form linear/nonlinear propagator or with a numerical integration scheme?
- What is the associated linearized dynamics model you will use to perform the EKF covariance update? Provide the state transition matrix $\mathbf{\Phi}$ and control input matrix $\mathbf{B}$.
- What sensors are available on-board your spacecraft, and what kinds of measurements do they provide (e.g. GNSS, radio frequency, optical, etc.)?
- What is the nonlinear model which relates the state to the sensor measurements?
- What is the associated sensitivity matrix $\mathbf{H}$ which you will use to perform the EKF update step?

Part 2. Navigation System Implementation

Now that you have set up foundation of your extended Kalman filter, you will implement an estimator in MATLAB/Simulink. Be sure to perform the following steps:

- Generate ground truth signals using a clearly-described formation propagator.
- Consider two types of measurements:
  1. Corrupt the ground truth state of interest with some Gaussian noise and feed this to the EKF (i.e. $y_t = Ix_t + v_t$, where v_t is zero-mean white noise). Use this simplified set of measurements to validate your filter's base performance.
  2. Generate representative measurements according to your mission's sensors using the nonlinear measurement model from Part 1.

  In both of the above cases, be sure to use apply reasonable levels of noise for your measurements based on the accuracy of your sensors.
  Note that you can generate noise as a zero-mean Gaussian random vector with preselected standard deviation in MATLAB as `sqrtm(sigma)*randn(n, 1)`, where `sigma` is a covariance matrix, and `n` is your desired vector size.

- Set an initial estimate and covariance, x_0 and P_0. The estimate should be close to, but not exactly equal to the true initial state. The covariance can be set to a diagonal matrix, with elements equal to the variance of your state parameters.
- Define the process and measurement noise covariances, $\mathbf{Q}$ and $\mathbf{R}$, which represent the uncertainty in the dynamics and measurement models, respectively. You may wish to define $\mathbf{Q}$ similar to P_0, but much smaller. Meanwhile $\mathbf{R}$ may be a diagonal matrices with elements equal to the variance of each measurement.
- Combine all of the previous parameters and mechanize your extended Kalman filter. Recall the steps involved in a single iteration of the EKF:
  1. Predict the evolution of the state and covariance with your dynamics model and state transition matrix.
  2. Generate a pre-fit residual by comparing the expected measurement against the true measurement.
  3. Update the prediction with the EKF measurement update step.
  4. Generate a post-fit residual by comparing the updated expected measurement against the true measurement.
- Illustrate the performance of the filter by plotting the following parameters: true, measured, and estimated states, estimation error (difference between true and estimated states), standard deviation of the estimate, and pre-fit and post-fit residuals.