AEM/ECE 566 Project 5

Ballistic Vehicle Altimetry System Design

Learning Objective

This project is intended to introduce the use of nonlinear Bayesian filters to estimate the altitude, vertical speed, and an unknown ballistic coefficient of a reentry vehicle using an altimeter and ballistic dynamics.

Sensor System

The continuous-time altitude dynamics model for a ballistic vehicle is given by

$$\begin{bmatrix} \dot{h} \\ \dot{s} \end{bmatrix} = \begin{bmatrix} s \\ \frac{\rho_0 s^2}{2C_b} \exp\left(\frac{-h}{h_\rho}\right) - g_0 \left(\frac{\bar{R}_E}{\bar{R}_E + h}\right)^2 \end{bmatrix} \tag{1}$$

where h is the altitude, s is the vertical speed, ρ_0 is the air density at mean sea level, h_ρ is the density reference altitude, g_0 is the standard acceleration due to gravity, $\bar{R}_E$ is the Earth's mean radius, and C_b is the ballistic coefficient. Here, the vertical acceleration, $\dot{s}$, is composed of a drag term and a gravity term for ballistic reentry. Using simple Euler integration for the time step Δt, the discrete-time altitude dynamics model for a ballistic vehicle is given by

$$\begin{bmatrix} h_k \\ s_k \end{bmatrix} = \begin{bmatrix} h_{k-1} + \Delta t s_{k-1} + w_{1,k-1} \\ s_{k-1} + \Delta t \left(\frac{\rho_0 s_{k-1}^2}{2C_b} \exp\left(\frac{-h_{k-1}}{h_\rho}\right) - g_0 \left(\frac{\bar{R}_E}{\bar{R}_E + h_{k-1}}\right)^2\right) + w_{2,k-1} \end{bmatrix} \tag{2}$$

where $\vec{w}_{k-1} = [w_{1,k-1} \;\; w_{2,k-1}]^T$ is additive zero-mean Gaussian white noise with covariance Q.

C_b can be modeled for a ballistic vehicle as

$$C_b = \frac{m}{S_A C_D} \tag{3}$$

where m is the vehicle mass, S_A is the cross-sectional area of the ballistic vehicle, and C_D is the drag coefficient. These parameters are often uncertain and vary with the atmospheric conditions during reentry. To jointly estimate the ballistic coefficient parameter with the altitude and vertical speed, one can form the augmented stochastic state-space system

$$\begin{bmatrix} h_k \\ s_k \\ C_{b,k} \end{bmatrix} = \begin{bmatrix} h_{k-1} + \Delta t s_{k-1} + w_{1,k-1} \\ s_{k-1} + \Delta t \left(\frac{\rho_0 s_{k-1}^2}{2C_B} \exp\left(\frac{-h_{k-1}}{h_\rho}\right) - g_0 \left(\frac{\bar{R}_E}{\bar{R}_E + h_{k-1}}\right)^2\right) + w_{2,k-1} \\ C_{b,k-1} + w_{3,k-1} \end{bmatrix} \tag{4}$$

where additive white noise has been added to the dynamics of the unknown parameter to allow for nonlinear Bayesian filtering to be implemented.

The radar model for the ballistic vehicle from a radar positioned at a horizontal distance d at mean sea level from the ballistic vehicle is given by

$$y_k = \sqrt{d^2 + h_k^2} + v_k \tag{5}$$

where y_k is the range and v_k is zero-mean white Gaussian noise with variance R.

Sensor Data

The .csv file provided contains by column:

1. the time $t = k\Delta t$,
2. the current range measurement, y_k,
3. the true altitude, h_k,
4. the true vertical velocity, s_k,
5. the true ballistic coefficient, $C_{b,k}$.

For this project assume:

- $\Delta t = 0.5$ s,
- $\rho_0 = 0.0765$ lb/ft^3,
- $g_0 = 32.2$ ft/s^2,
- $h_\rho = 30,000$ ft,
- $\bar{R}_E = 20,902,260$ ft,
- $d = 100,000$ ft,
- $Q = \text{diag}(10^2,\ 10^2,\ 0.05^2)$,
- $R = 100^2$ ft^2,
- $\hat{\vec{x}}_0 = [400,000 \ \ -2,000 \ \ 20]^T$ [ft, ft/s, lb/ft^2]^T, and
- $P_0 = \text{diag}(100^2,\ 10^2,\ 1^2)$ [ft^2, ft^2/s^2,]^T.

Project Assignment and Deliverables

Do: the following tasks in MATLAB or Python.

a) Implement an extended Kalman filter (EKF) to estimate the vehicle states, h and s, and the ballistic coefficient parameter, C_b.

- Plot the posterior state estimates versus the true states for the altitude.
- Plot the posterior state estimates versus the true states for the vertical speed.
- Plot the posterior state estimates versus the true states for the ballistic coefficient.

b) Implement a sigma-point Kalman filter (SPKF), e.g., an unscented Kalman filter (UKF), to estimate the vehicle states, h and s, and the ballistic coefficient parameter, C_b.
   - Plot the posterior state estimates versus the true states for the altitude.
   - Plot the posterior state estimates versus the true states for the vertical speed.
   - Plot the posterior state estimates versus the true states for the ballistic coefficient.

c) Implement a bootstrap particle filter (BPF) to estimate the vehicle states, h and s, and the ballistic coefficient parameter, C_b.
   - Plot the posterior state estimates versus the true states for the altitude.
   - Plot the posterior state estimates versus the true states for the vertical speed.
   - Plot the posterior state estimates versus the true states for the ballistic coefficient.

d) Comment on the accuracy and computational costs of each method.
   - Plot the posterior state estimate errors versus time for all three filters.

Deliver: in the Blackboard assignment, all files to run your MATLAB or Python script(s). There is no need to zip your files.