

UNIVERSITY OF COLORADO - BOULDER

STATISTICAL ESTIMATION OF DYNAMICAL SYSTEMS: ASEN5044

11/28/2025

---

## Cooperative Air-Ground Robot Localization Progress Report 1

---

*Author:*

Aaron Pineda

[AARON.PINEDA@COLORADO.EDU](mailto:AARON.PINEDA@COLORADO.EDU)

*Author:*

Jeremy Aubert

[JEREMY.AUBERT@COLORADO.EDU](mailto:JEREMY.AUBERT@COLORADO.EDU)

*Professor:*

Professor KHOSRO  
GHOBADI FAR

*Author:*

Landry Matthews

[LANDRY.MATTHEWS@COLORADO.EDU](mailto:LANDRY.MATTHEWS@COLORADO.EDU)



Ann and H.J. Smead  
Aerospace Engineering Sciences

UNIVERSITY OF COLORADO BOULDER

## I. Introduction

Accurate localization and navigation remain difficult for autonomous robots in environments where GPS is unreliable or compromised. Cooperative localization provides a way to improve robustness by allowing vehicles to share relative measurements with teammates. This project examines a system with a ground vehicle (UGV) and an aerial vehicle (UAV) that exchange relative measurements during an encounter. The UAV maintains GPS access, while the UGV relies on cooperative tracking. Although the broader framework supports decentralized estimation, this report focuses on developing a centralized estimator with nonlinear motion models as a baseline. The initial phase presented here implements both Linear and Extended Kalman filters to estimate the states of the two robots.

## II. Part I: Deterministic System Analysis

### A. Linearizing the CT model

Given the initial equations of the system we found our CT jacobians, which are:

$$A(x, u) = \begin{bmatrix} 0 & 0 & -v_g \sin \theta_g & 0 & 0 & 0 \\ 0 & 0 & v_g \cos \theta_g & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -v_a \sin \theta_a \\ 0 & 0 & 0 & 0 & 0 & v_a \cos \theta_a \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

,

$$B(x, u) = \begin{bmatrix} \cos \theta_g & 0 & 0 & 0 \\ \sin \theta_g & 0 & 0 & 0 \\ \frac{1}{L} \tan \phi_g & \frac{v_g}{L} \sec^2 \phi_g & 0 & 0 \\ 0 & 0 & \cos \theta_a & 0 \\ 0 & 0 & \sin \theta_a & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

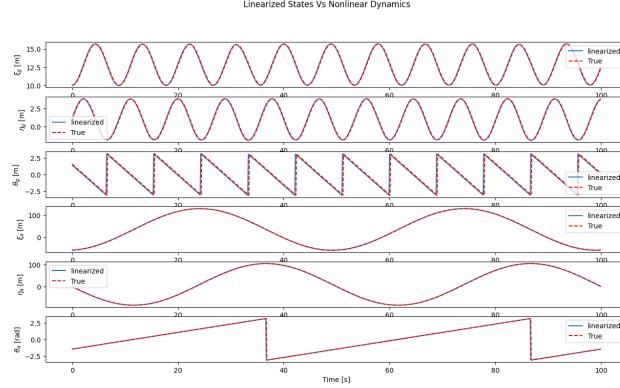
$$H(x) = \begin{bmatrix} \frac{\eta_a - \eta_g}{(\xi_a - \xi_g)^2(1 + \frac{\eta_a - \eta_g}{\xi_a - \xi_g})} & -\frac{1}{(\xi_a - \xi_g)(1 + \frac{\eta_a - \eta_g}{\xi_a - \xi_g})} & -1 & -\frac{\eta_a - \eta_g}{(\xi_a - \xi_g)^2(1 + \frac{\eta_a - \eta_g}{\xi_a - \xi_g})} & \frac{1}{(\xi_a - \xi_g)(1 + \frac{\eta_a - \eta_g}{\xi_a - \xi_g})} & 0 \\ -\frac{2(\xi_a - \xi_g)}{\sqrt{(\xi_a - \xi_g)^2 + (\eta_a - \eta_g)^2}} & -\frac{2(\eta_a - \eta_g)}{(\xi_a - \xi_g)^2 + (\eta_a - \eta_g)^2} & 0 & \frac{2(\xi_a - \xi_g)}{(\xi_a - \xi_g)^2 + (\eta_a - \eta_g)^2} & \frac{2(\eta_a - \eta_g)}{(\xi_a - \xi_g)^2 + (\eta_a - \eta_g)^2} & 0 \\ -\frac{\eta_g - \eta_a}{\xi_g - \xi_a}(1 + \frac{\eta_g - \eta_a}{\xi_g - \xi_a}) & \frac{1}{\xi_g - \xi_a}(1 + \frac{\eta_g - \eta_a}{\xi_g - \xi_a}) & 0 & \frac{\eta_g - \eta_a}{\xi_g - \xi_a}(1 + \frac{\eta_g - \eta_a}{\xi_g - \xi_a}) & \frac{1}{\xi_g - \xi_a}(1 + \frac{\eta_g - \eta_a}{\xi_g - \xi_a}) & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

however, because the system is time varying the jacobians change with every time step. Therefore our approach was to determine the CT dynamics of the system,  $\tilde{A}$ ,  $\tilde{B}$  and  $\tilde{C}$ , then convert to DT dynamics using the Forward Euler method to find  $\tilde{F}$  and  $\tilde{G}$ .  $\tilde{H}$  is obtained from linearizing the nonlinear measurement function  $h(x)$ . In the linear case  $\tilde{H}$  reduces to  $\tilde{C}$ . The equations for  $\tilde{F}$  and  $\tilde{G}$  were derived in lecture to be  $\tilde{F} = I + \Delta t \tilde{A}$  and  $\tilde{G} = \Delta t \tilde{B}$ . After converting the system to DT, we then linearized around a nominal trajectory at each time step. Since the nominal trajectory was different at each timestep of the system, the nonlinear equations needed to be integrated over the desired range of 100 seconds using solve IVP from the python library scipy. For this we used the initial nominal state condition that was provided of  $x_0 = [10, 0, \frac{\pi}{2}, -60, 0, -\frac{\pi}{2}]$ . For each timestep the computed nominal trajectory was used to evaluate each of the state matrices and determine the future steps perturbation through the equation  $\delta x(k+1) \approx \tilde{F}_{nom}(k)\delta x(k)$ . For this part contributions from noise and control inputs were ignored for perturbation vectors.

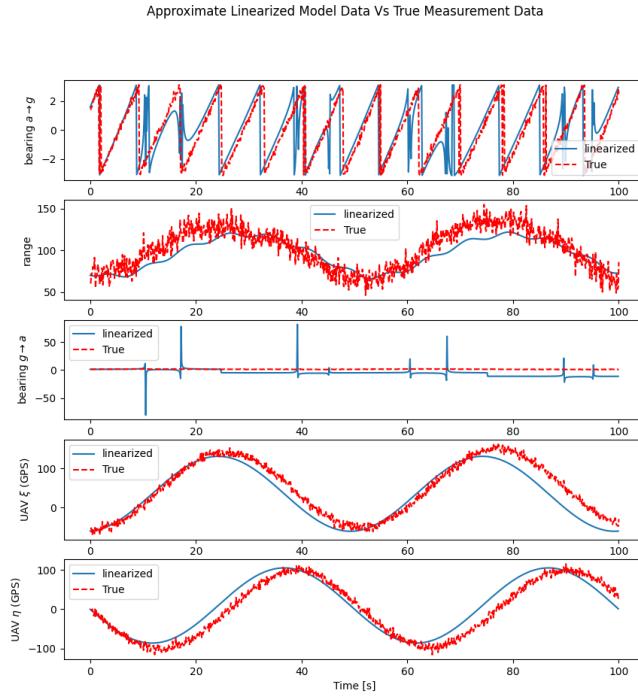
### B. Simulating Linearized DT Dynamics

One thing to be noted is that the angles for the UGV and UAV had to be bounded within the specified limitations, and were bounded as such in the script to generate the correct plots. Firstly, the systems nonlinear dynamics were plotted

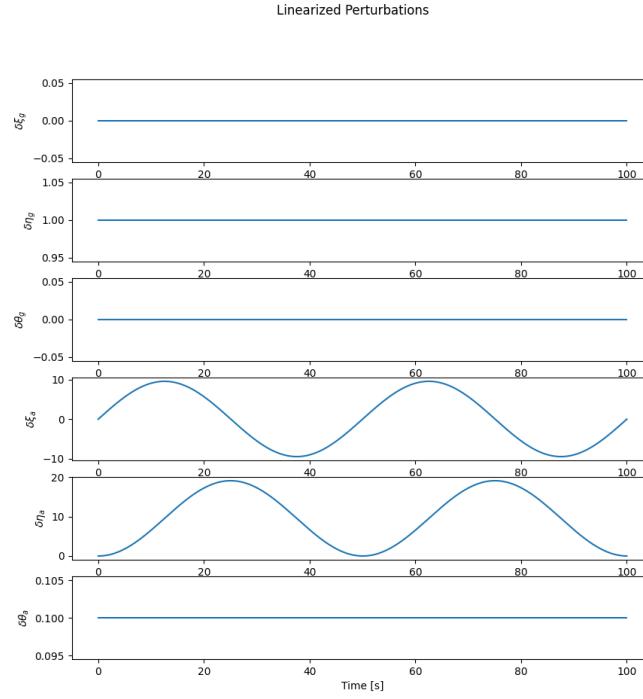
against the estimated linearized dynamics. The initial perturbation used was  $x_0 = [0; 1; 0; 0; 0; 0.1]$  with nominal control inputs and no process or measurement noise in the simulation. The nonlinear states were generated from solve IVP for 100 seconds(1000 time steps) which was overlayed with the simulated linearized dynamics. Following this plot is the simulated measurement data and true measurement data, following finally with the perturbation dynamics.



**Fig. 1 Modeled Dynamics.**



**Fig. 2 Measurements.**



**Fig. 3 Perturbation Dynamics.**

### III. Contributions

Each team member is individually working through the project to allow us to apply what we have learned and attempt to solidify our understanding of the course concepts. The code and images in this report come from Aaron's branch in the git repo. <https://github.com/Aaron2014/ASEN5044-Cooperative-Air-Ground-Robot-Localization>