Tucker Guen - Project 1

Project Overview

Although all files are well commented and their purposes should be apparent, here is an overview of the important files and their purposes:

- config_ws.m - Sets up the project workspace, adding variables for the map, initial states and beliefs, motion command and measurement lists, and evaluates the true robot motion with no motion error.
- EKF.m - Implementation of the extended kalman filter for a single timestep. Takes in the previous state belief, a motion command, a measurement, a map, and error parameters
- EKF_main.m - Runs the extended kalman filter of all timesteps and plots the graph of motion, pre-update uncertainty, and post-update uncerainty
- particle_filter.m - Implementation of the particle filter for a single timestep. Takes in the previous timestep particle set, a motion command, a measurement, a map, and error parameters.
- PF_main.m - Runs the particle filter for all timesteps and plots the pre-update and post-update particle sets.

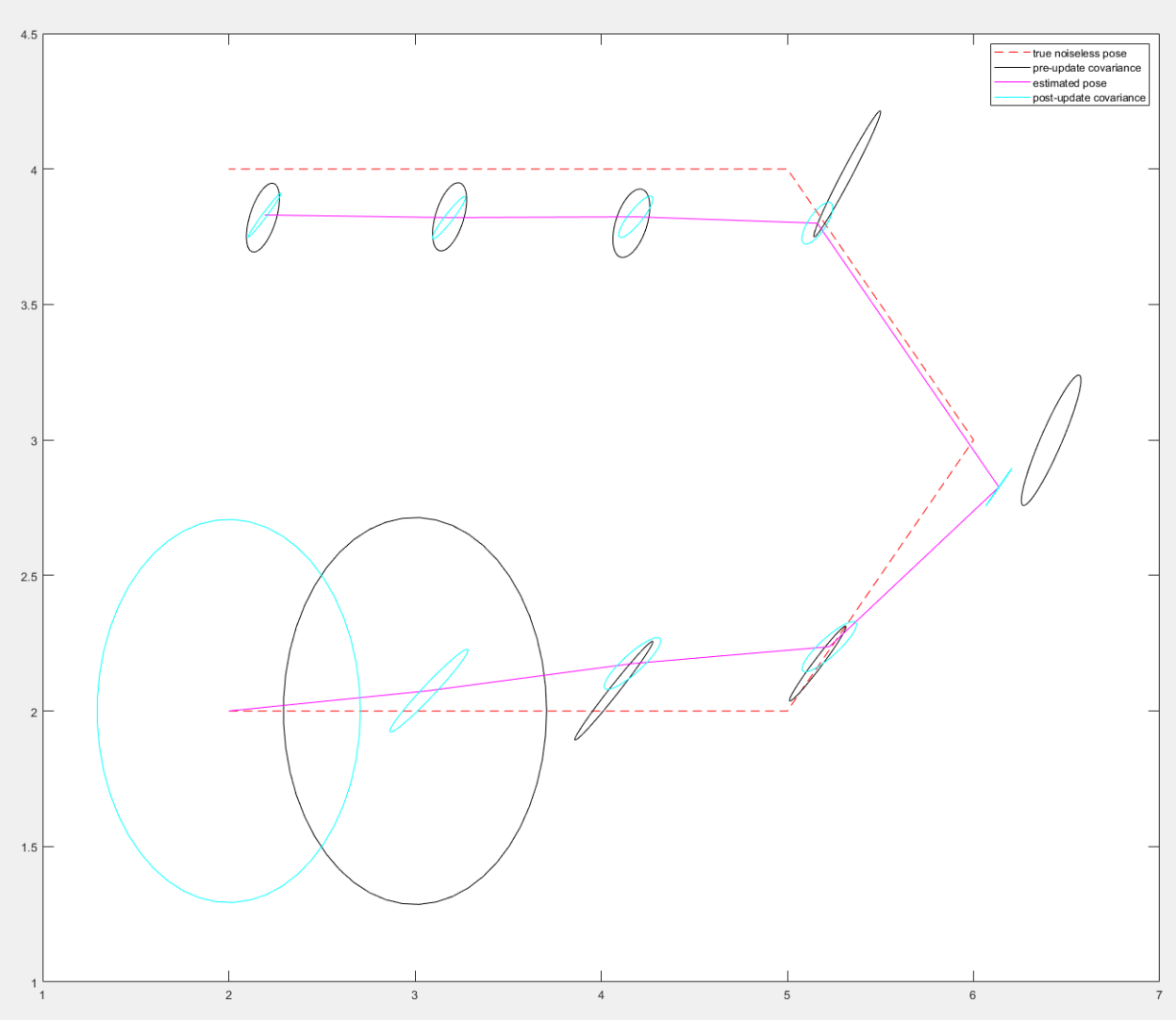
All other files are either classes that serve as abstract data types for storing aspects of the state space, or are scripts with clear purpose.

- Note: plot_gaussian_ellipsoid.m was sourced from: https://www.mathworks.com/matlabcentral/fileexchange/16543-plot_gaussian_ellipsoid

Extended Kalman Filter

The initial covariance was picked to be a 3×3 identity matrix. The first two circles are of somewhat arbitrary size.

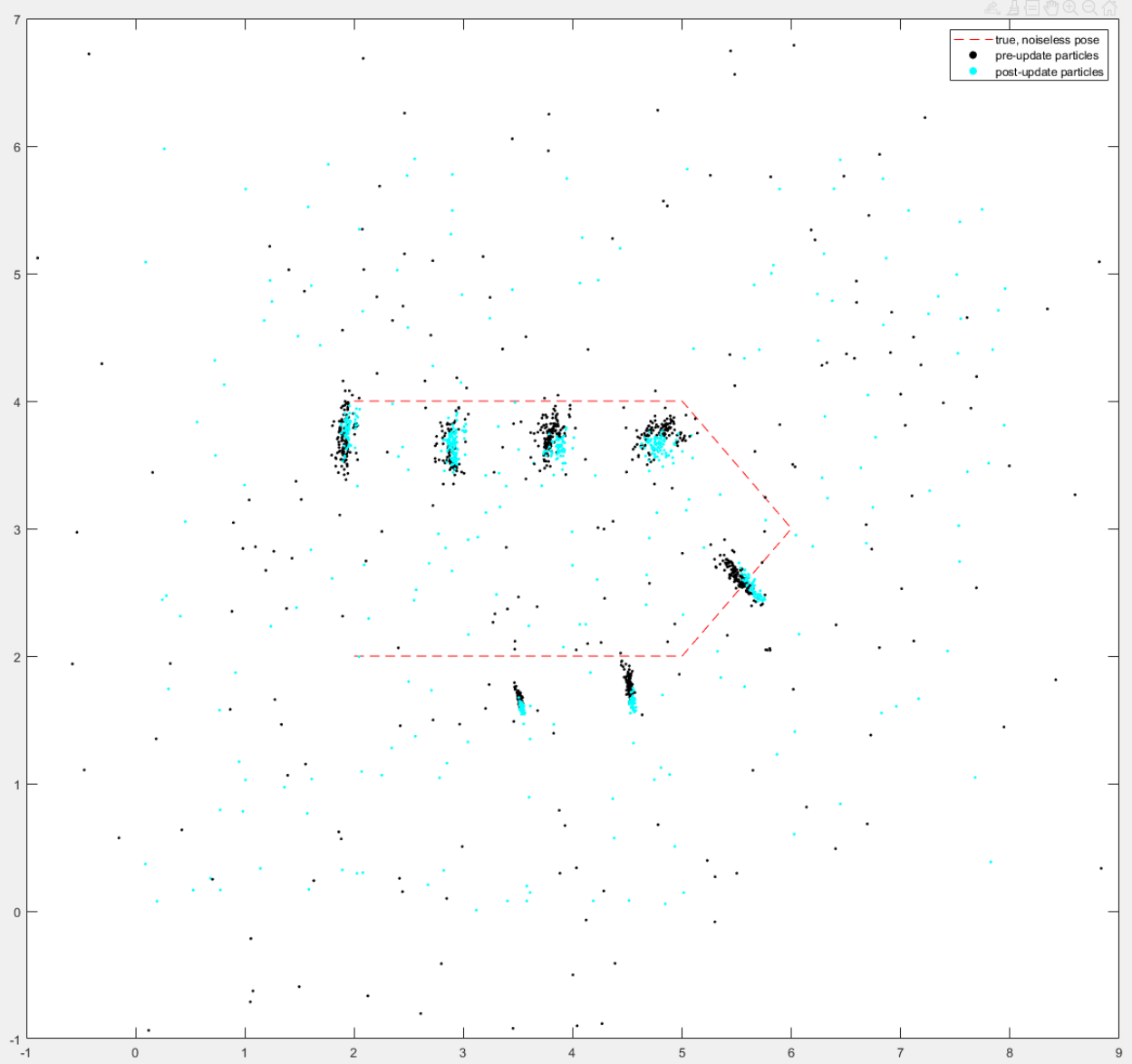
- Legend (also labeled on figure)
  - Orange dashed line: True robot motion using velocity motion model with same input commands and no motion uncertainty $(\alpha = [0, ..., 0])$
  - Magenta solid line: Noisy robot motion computed during each step of the EKF
  - Black Circles: The pose covariance ellipse before the measurement update
  - Cyan Circles: The pose covariance ellipse after the measurement update

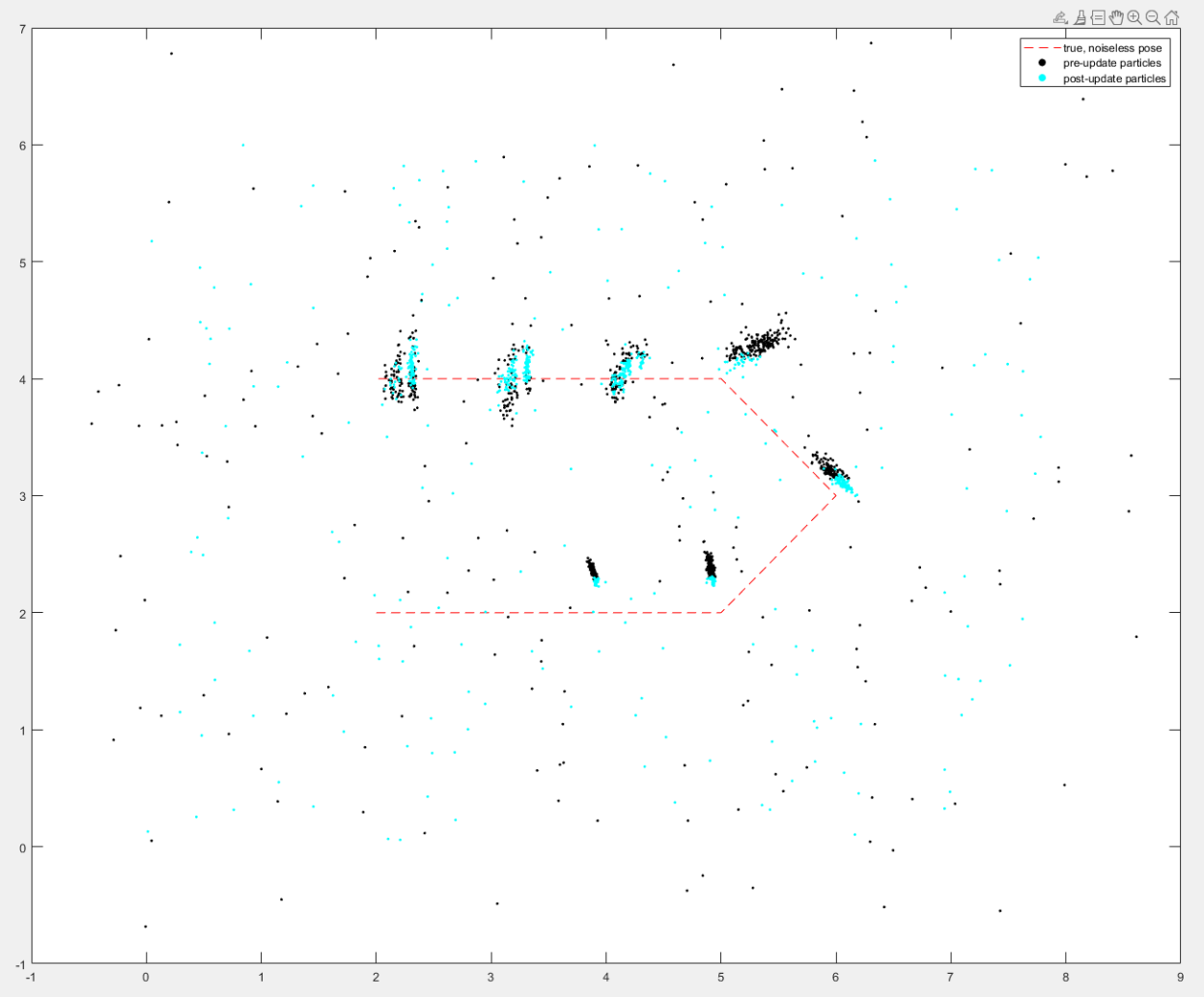

true noiseless pose
pre-update covariance
estimated pose
post-update covariance
4.5
4
3.5
3
2.5
2
1.5
1
1
2
3
4
5
6
7

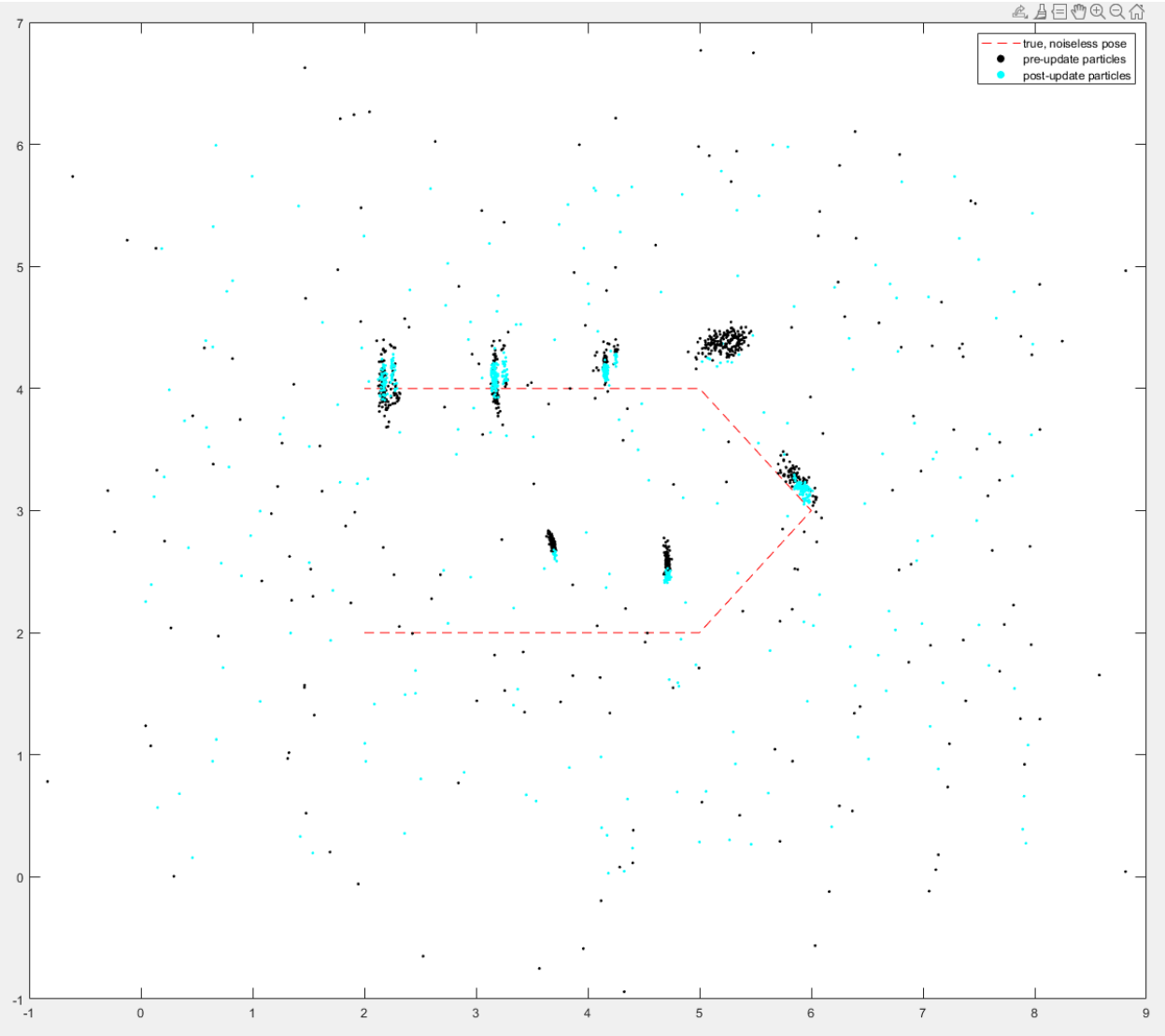
Particle Filter

- Legend:
  - Orange dashed line: True robot motion using velocity motion model with same input commands and no motion uncertainty ($\alpha = [0, ..., 0]$)
  - Black Dots: The set of particles before measurement update (resampling)
  - Cyan Dots: The set if particles after measurement update

Due to the randomness of resampling, the results of the particle filter can vary, but here are a few good runs that converge well and demonstrate resampling.


true, noiseless pose
pre-update particles
post-update particles
7
6
5
4
3
2
1
0
-1
-1
0
1
2
3
4
5
6
7
8
9


true, noiseless pose
pre-update particles
post-update particles
7
6
5
4
3
2
1
0
-1
-1
0
1
2
3
4
5
6
7
8
9


true, noiseless pose
pre-update particles
post-update particles
7
6
5
4
3
2
1
0
-1
-1
0
1
2
3
4
5
6
7
8
9