Project: Estimation

Project: Estimation

Writeup / README

1. Provide a Writeup / README that includes all the rubric points and how you addressed each one. You can submit your writeup as markdown or pdf.

You're reading it! Below I describe how I addressed each rubric point and where in my code each point is handled.

Build estimator

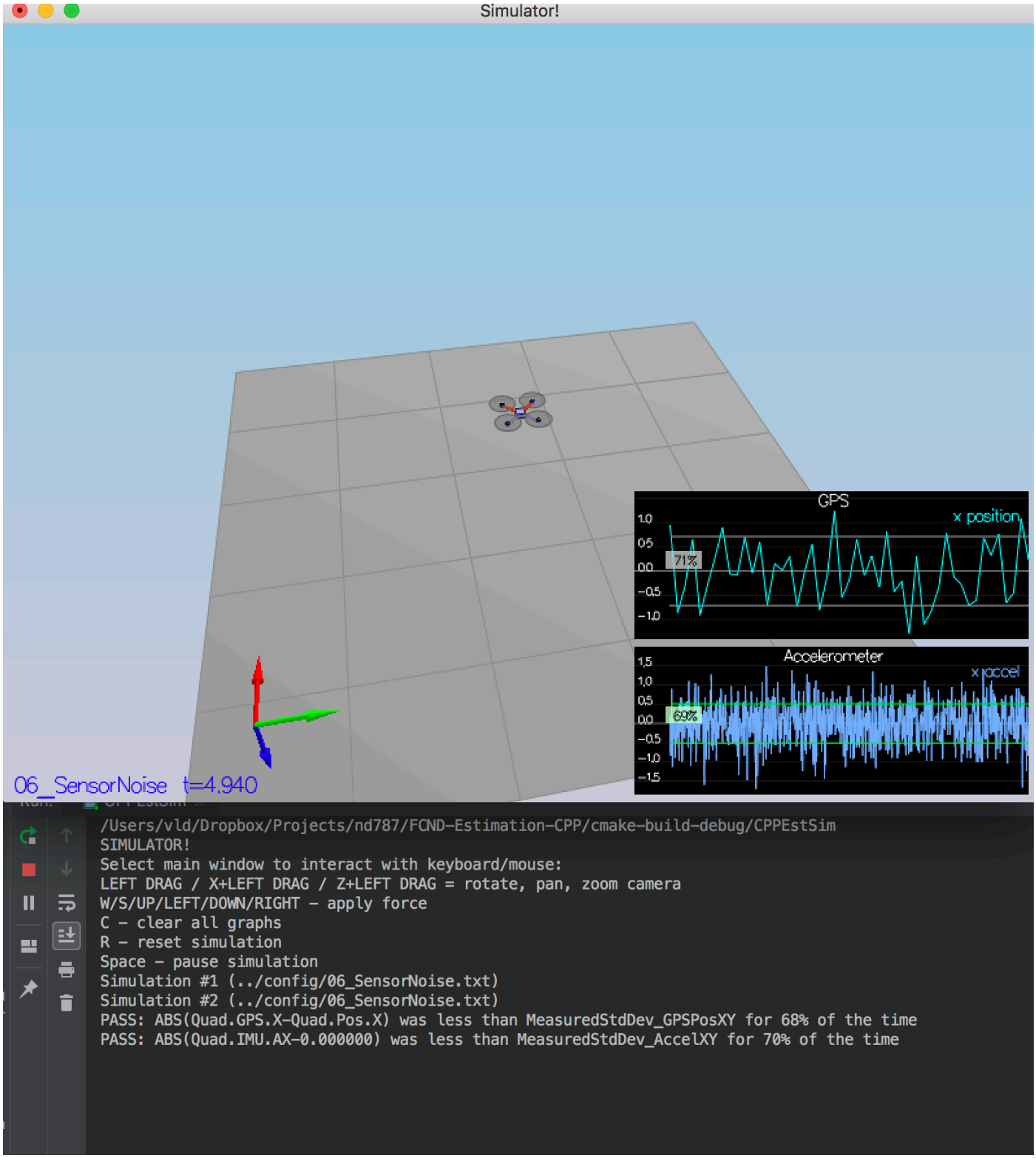
The estimator will be build in tasks:

Step 1: Sensor Noise
Step 2: Attitude Estimation
Step 3: Prediction Step
Step 4: Magnetometer Update
Step 5: Closed Loop + GPS Update
Step 6: Adding Your Controller

Step 1: Sensor Noise

I used the instructions from `README.md` to generate the timeseies. Then I calculated their standard deviation using `numpy.std`. The results I seved to `config/6_Sensornoise.txt` were:

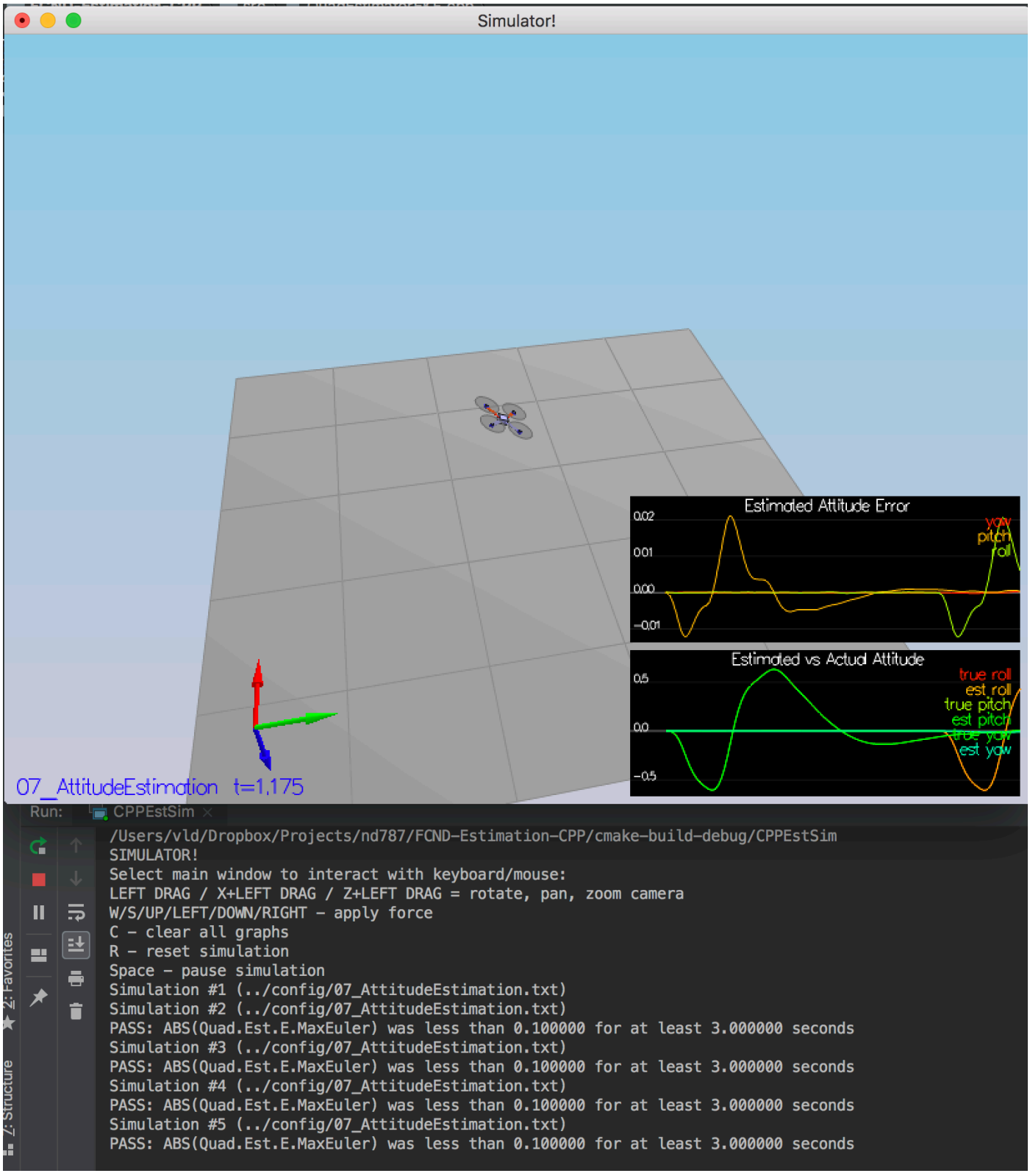
```
MeasuredStdDev_GPSPosXY = 0.71
MeasuredStdDev_AccelXY = 0.51
```



The numbers I've got were close to ones from `config/SimulatedSensors.txt` (0.7 and 0.5).

Step 2: Attitude Estimation

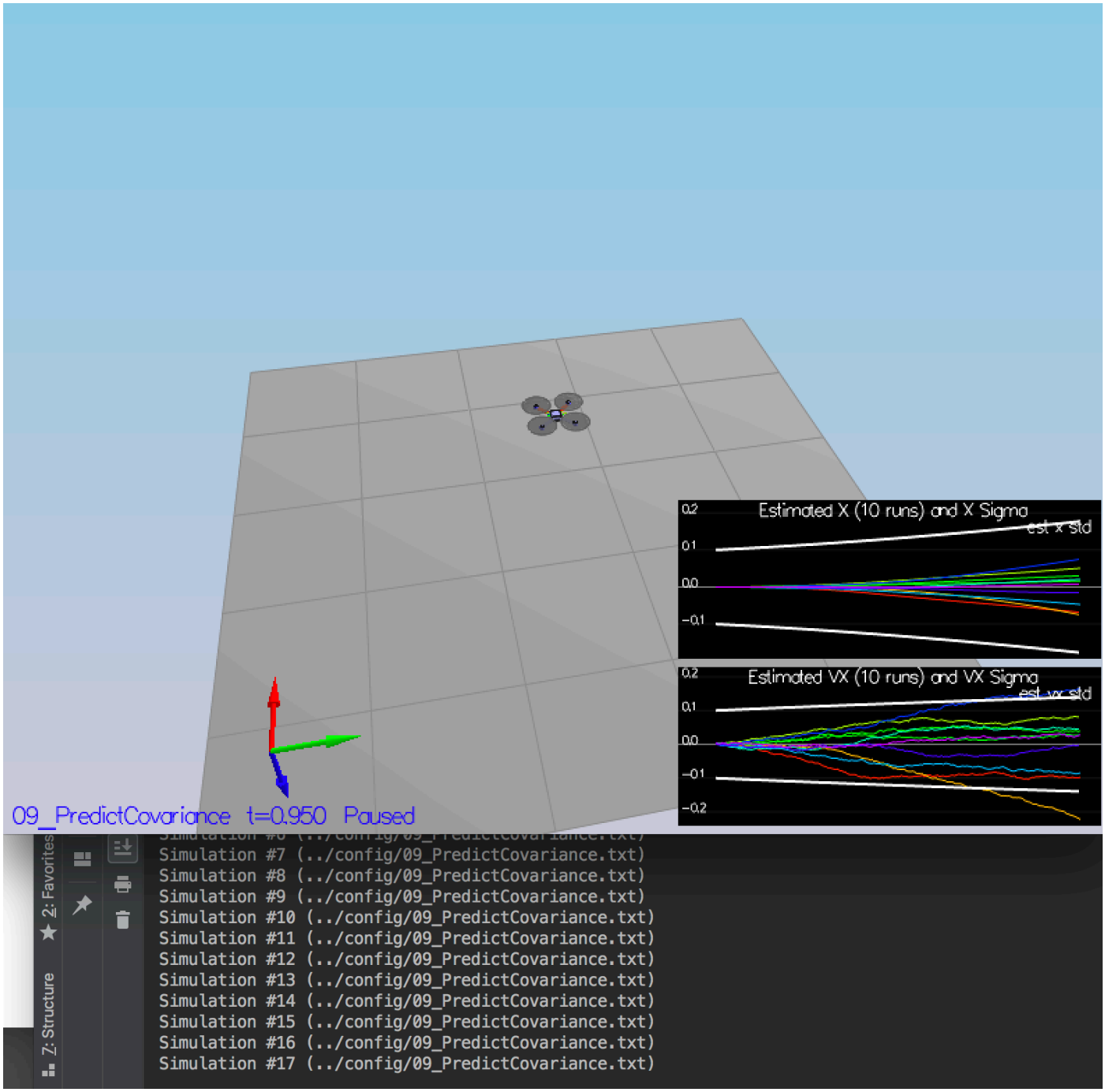
I was able to reduce the errors in the estimated attitude (Euler Angles) via implementing a better rate gyro attitude integration scheme in `UpdateFromIMU()` as it was advised.



Step 3: Prediction Step

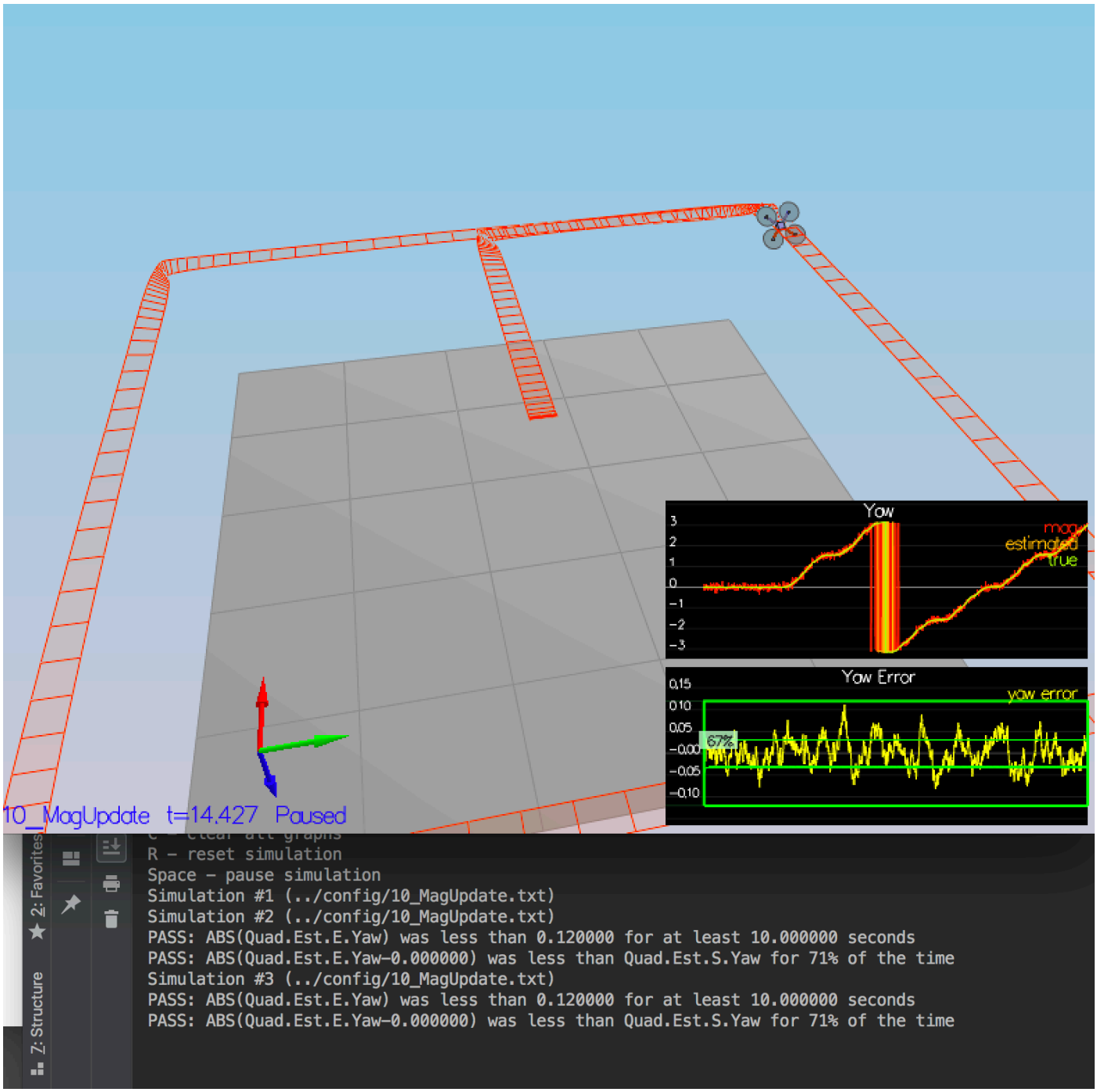
In this step, I implemented the required prediction steps for the estimator. For that, I modified the `Predict()`, `PredictState()` and `GetRbgPrime()` functions.

Also, I tuned the `QPosXYStd` and `QVelXYStd` parameters. My final Predict Covariance plots are below:



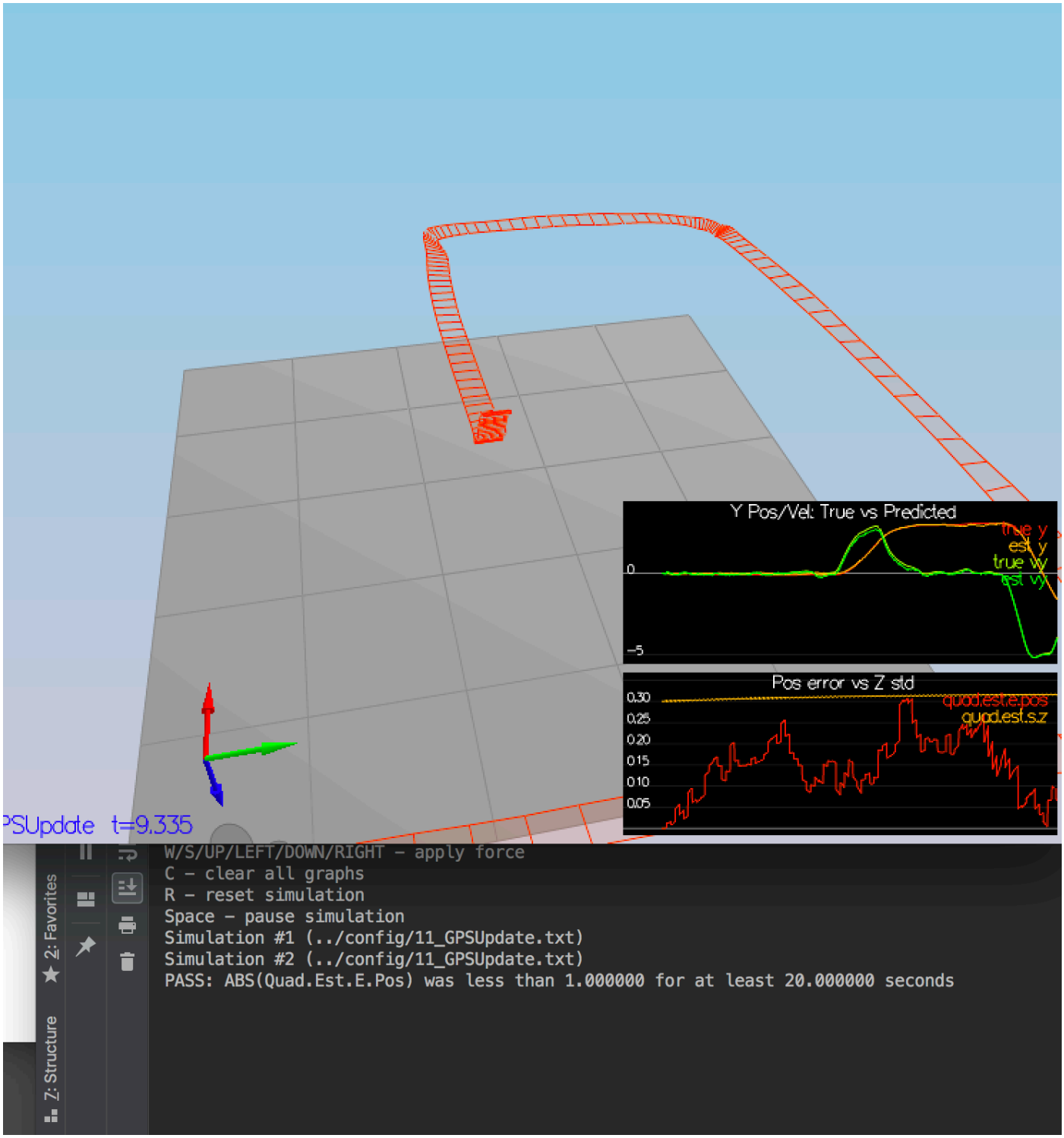
Step 4: Magnetometer Update

This was a step to include magnetometer values into the state. I implemented it in the `UpdateFromMag()` function after chosing a suitable `QYawStd` parameter:



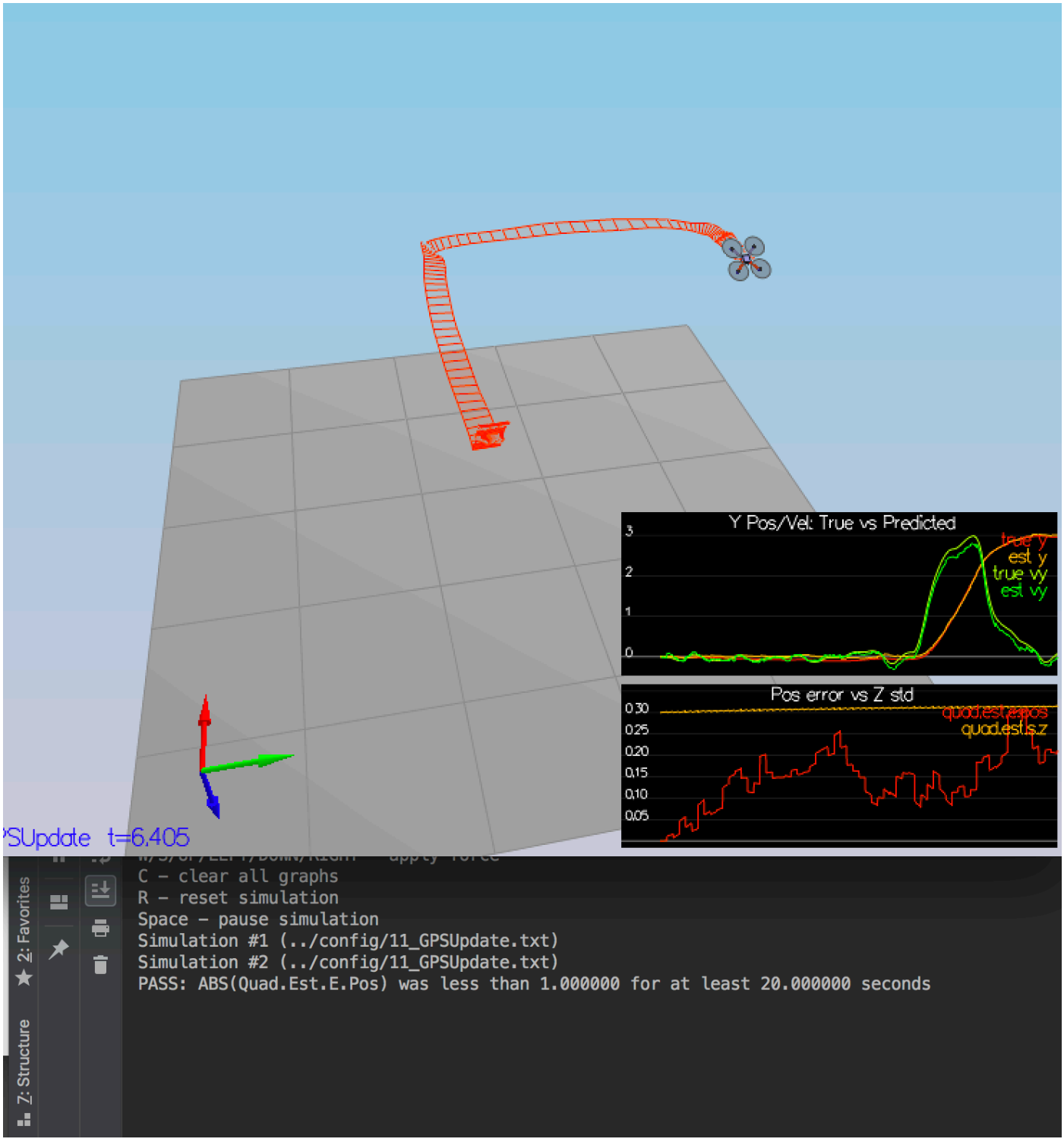
Step 5: Closed Loop + GPS Update

I has to implement the EKF GPS update in `QuadEstimatorEKF.txt` in this step. It was no required to update any other parameters in `QuadEstimatorEKF.txt` to reach the required objective:



Step 6: Adding Your Controller

I copied the existing implementation of `QuadController.cpp` from the previous project and just had to follow the advice on deduning the controller a bit. Adjusting position parameters was enough to meet the objective:



C - clear all graphs
R - reset simulation
Space - pause simulation
Simulation #1 (../config/11_GPSUpdate.txt)
Simulation #2 (../config/11_GPSUpdate.txt)
PASS: ABS(Quad.Est.E.Pos) was less than 1.000000 for at least 20.000000 seconds

2: Favorites
7: Structure