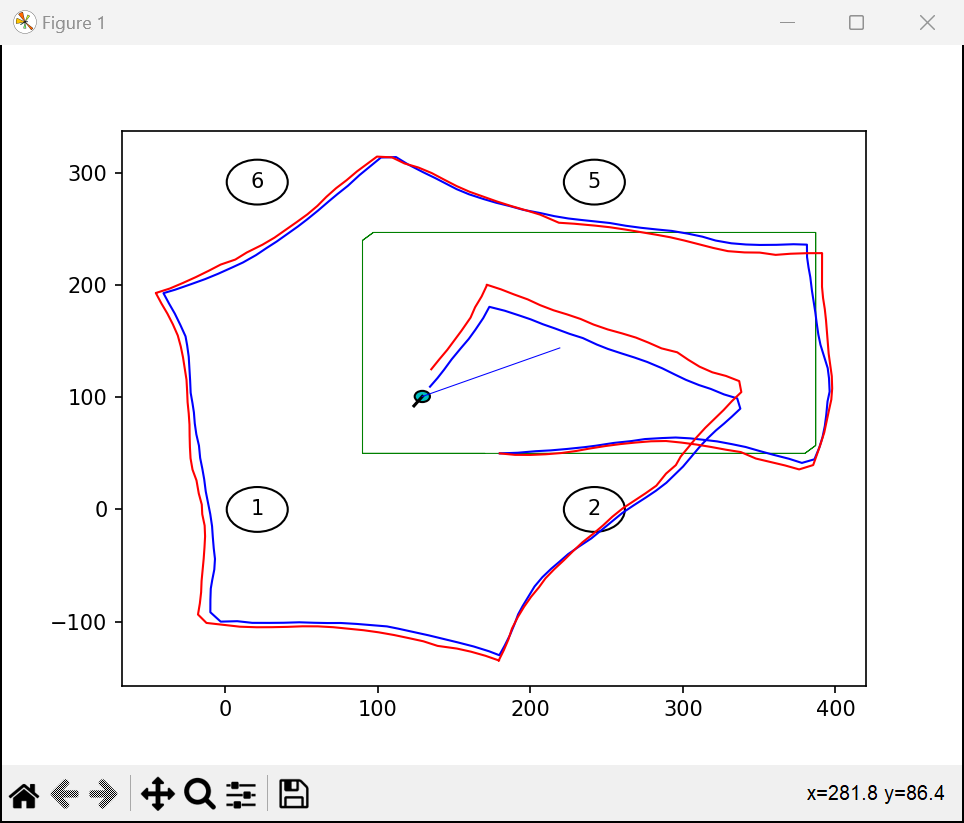
**Extended Kalman Filter Plots, Results and Observations**

Result with $ python localization.py ekf --seed 0:



Mean position error: 8.141191137920284

Mean Mahalanobis error: 4.525110065340756

ANEES: 1.5083700217802518

Definition of Mean Position Error:

Mean Position Error: Average of the distance between a point in the predicted path of the robot to the corresponding point in its actual path.

Full Form of ANEES:

Averaged normalised estimation error squared

Graphs:

(The complete data for the below plotted graphs can be found in the .xlsx file named ‘EKF Results’.)

Graph for Data factor = Filter factor = r

Graphs for Data factor = 1, Filter factor = r

(Since I have used my own Q matrix and haven not used the Q matrix defined in soccer\_field.py, thus for altering data factor for the below graph, I manually multiplied the Q in my code by 1/r (as the cmd multiplied the general alpha and beta by r.))

Observations:

When both, data factor and filter factor are r, then on increasing r exponentially, mean position error also increases rapidly.

In this case, the amount of uncertainty (covariance) is increase/decreased by the same factor for both, the data and the filter. Since error increase in both, thus, mean position error should also increase approximately by the same factor r.

When data factor is kept 1 and filter factor is multiplied by r, then on increasing r exponentially, the trend seen from the plotted graph is that the mean position nearly stays constant, but ANEES increases sharply with r. This maybe so because the data becomes more dependent on measurement due to a higher Kalman Gain.