COL 864: Planning and Estimation for Autonomous Systems

Assignment 1

Report submitted by

Ankit Garg **Karan Mittal**
2017EE10439 **2017ME20671**

INDIAN INSTITUTE OF TECHNOLOGY DELHI

24th March 2021

Contents

Chapter 1

Discrete State Space Problem

An aerial vehicle is flying at a constant height while surveying an area with dimensions 30m $\times$ 30m. Uniform discretization is assumed for the region with grid cells of size 1m $\times$ 1m. At each time step, the robot can execute one of four actions: up, down, left or right to an adjacent grid cell, selected with likelihoods 0.4, 0.1, 0.2 and 0.3 respectively. The area is instrumented with noisy sensors that report the discrete presence or absence of a target. The sensors are located at grid cells: (8, 15),(15, 15),(22, 15),(15, 22) where (i, j) indicates the index along x and y axes.

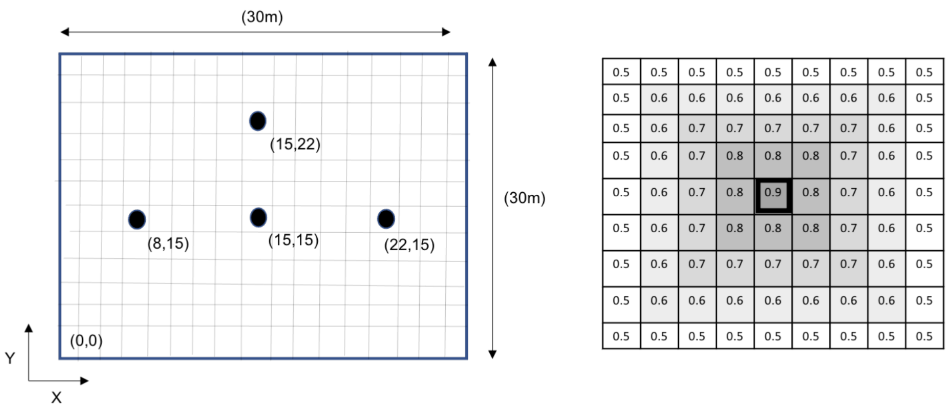

Figure 1.1: Discrete state space & sensor likelihood

1.1 Robot's Motion Simulation

We have simulated the Robot's Motion as seen in Figure 1.2 and Figure 1.3 by sampling an action from the given set of actions {up,down,left,right} according to the given likelihood. We have done this for 25 time steps and recorded the readings in Figure 1.4 of all the sensors at each time step. We have taken our starting point as (15,12).

Coordinates of the Sensors:

Sensor 1 : (8,15)

Sensor 2 : (15,15)

Sensor 3 : (15,22)

Sensor 4 : (22,15)

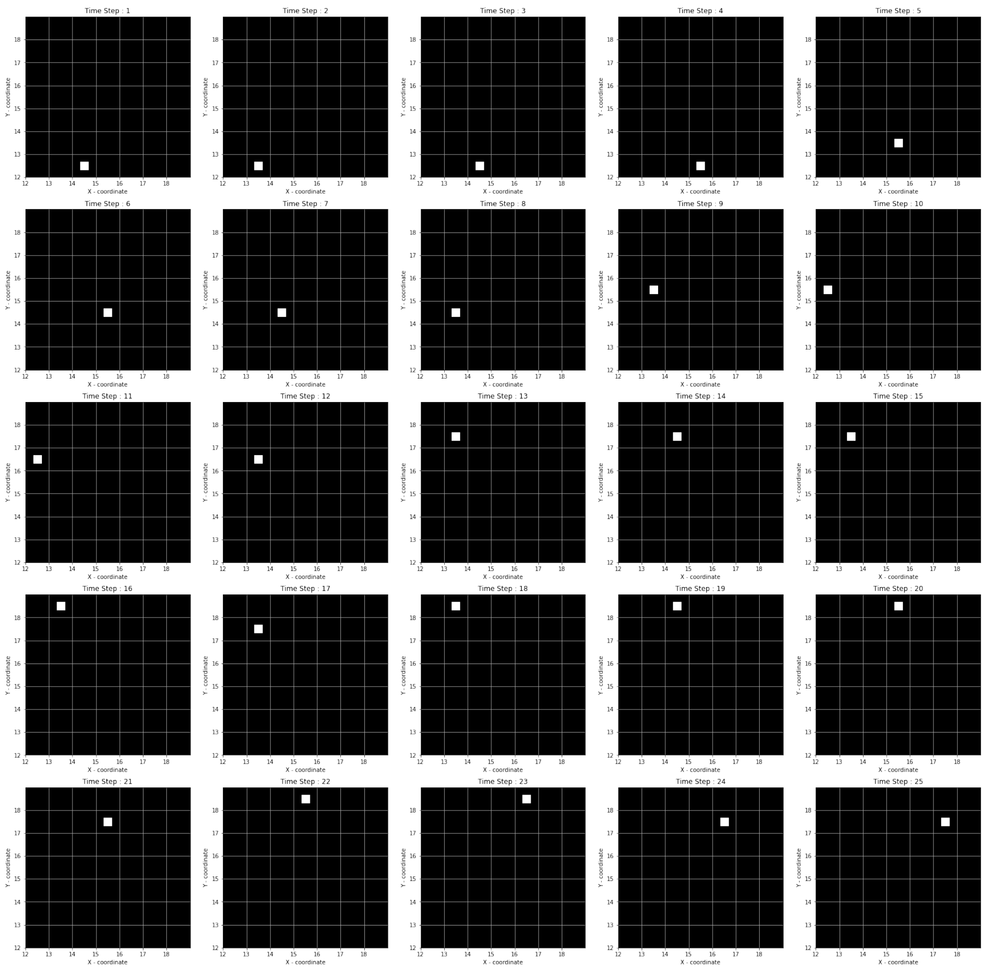

Figure 1.2: Simulated Path from t = 1 to t = 25

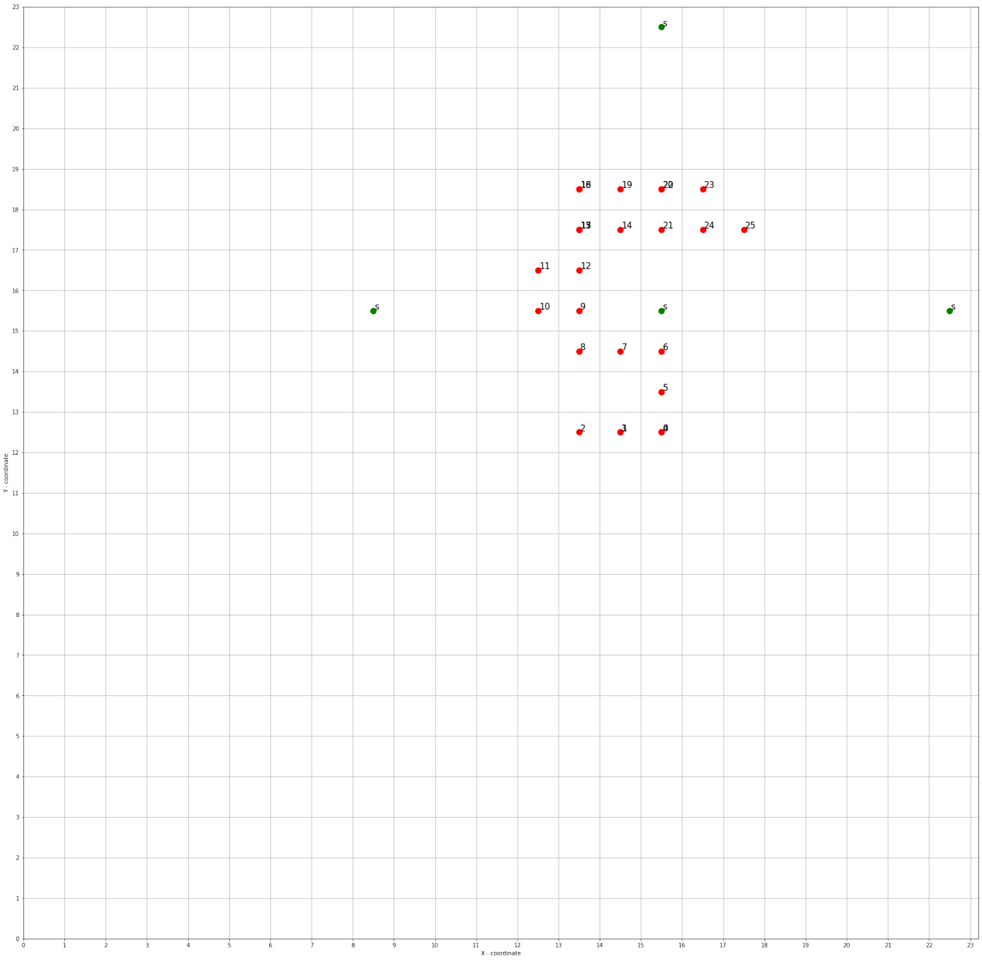

Figure 1.3: Path followed by the agent on the grid and location of sensors

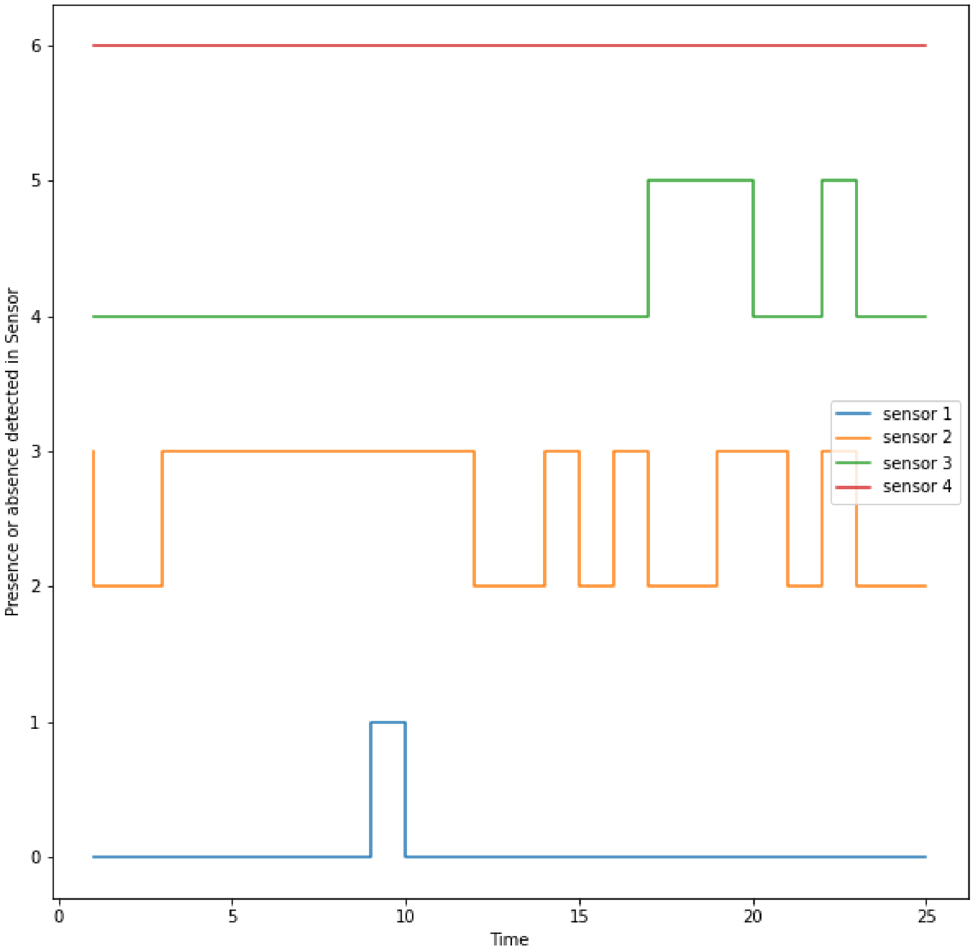

Figure 1.4: Sensor Recordings

1.2 Belief of Robot's current position

Calculated the belief as seen in Figure 1.5 of Robot's position at all time steps. In the heat map we can observe that there is more uncertainty in the initial part of the path. This is because with time we have more sensor observations and are able to estimate the location better. Estimated Locations were calculated by taking the argmax over Belief of all states for each time step. Estimated and Ground Truth locations are plotted in Figure 1.6. Even in the log likelihood plot in Figure 1.7 we observe that the peak is flatter in the starting and with time it becomes sharper.

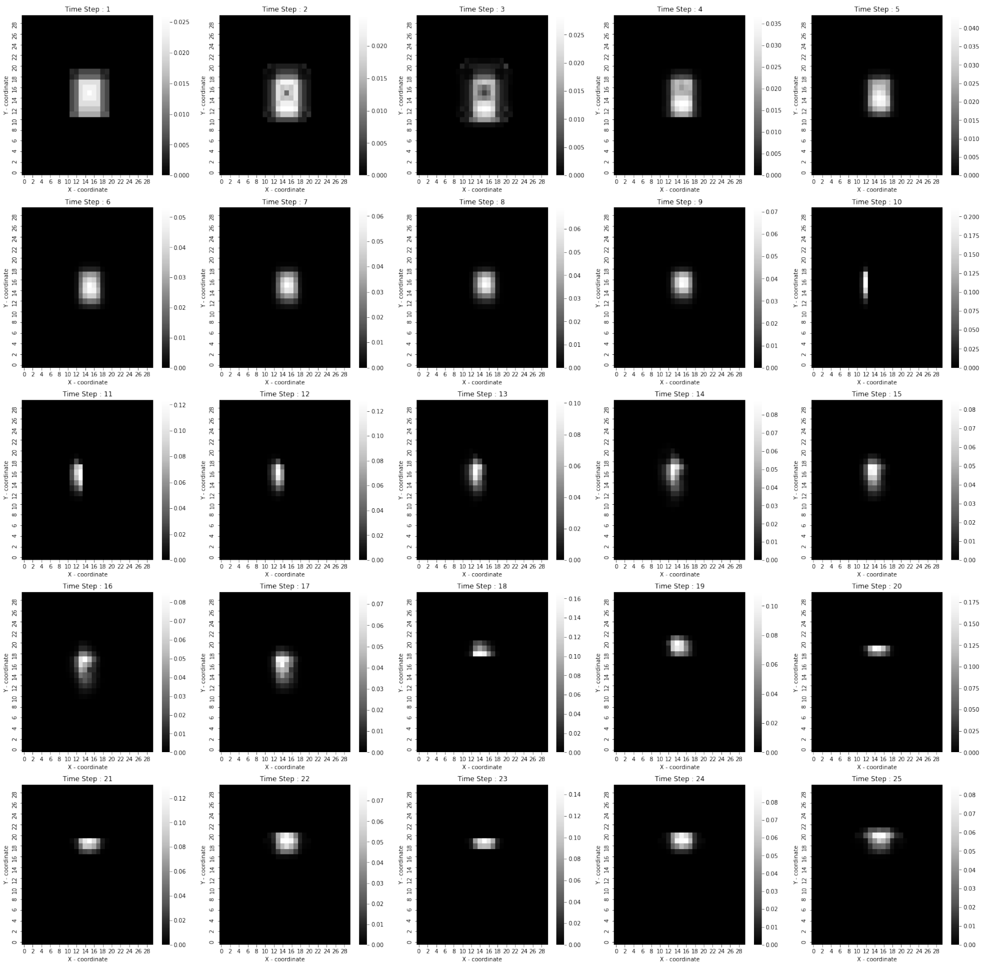

Figure 1.5: Belief Heat Map t=1 to t=25

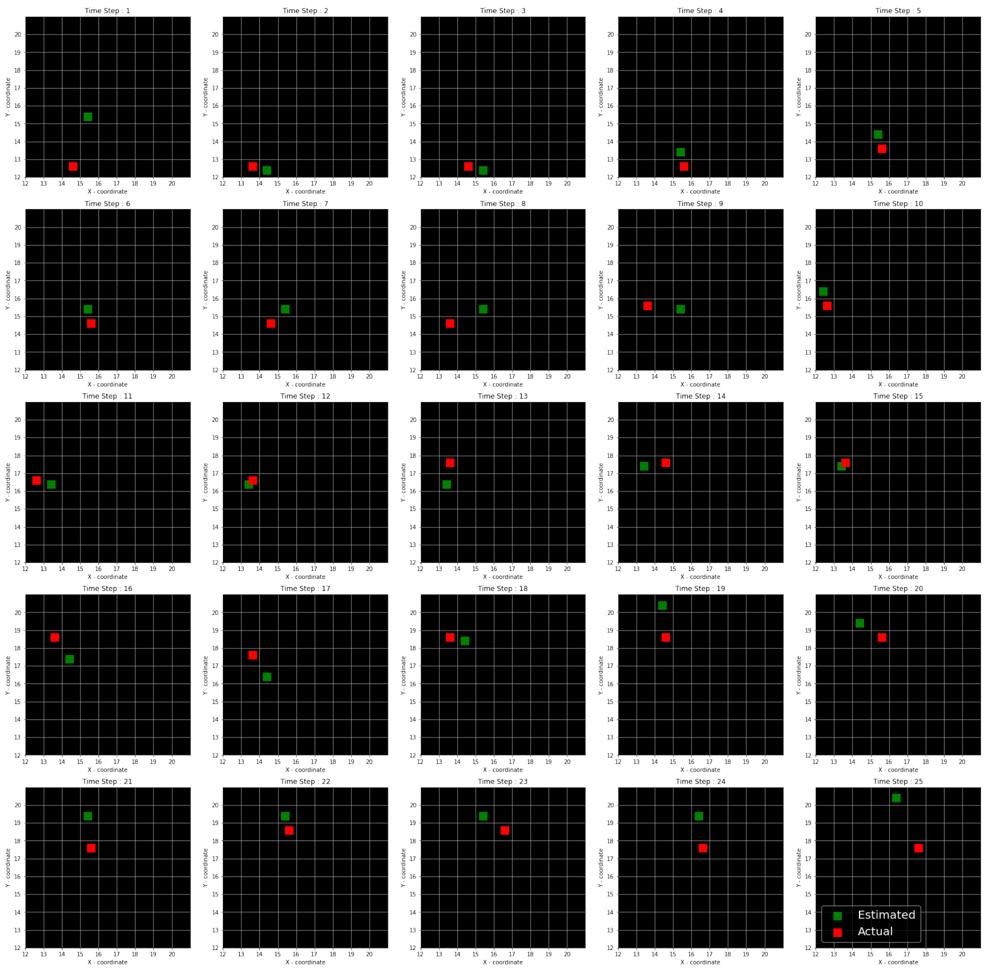

Figure 1.6: Estimated and Ground Truth Locations: t=1 to t=25

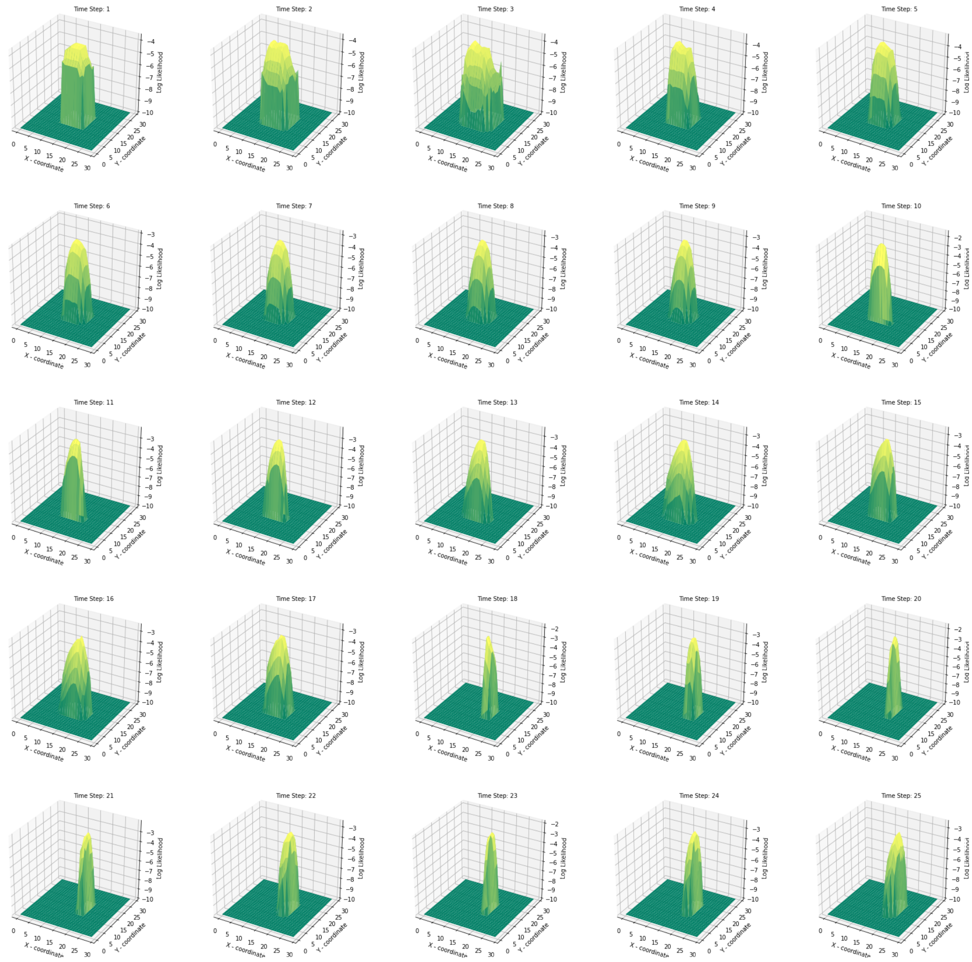

Figure 1.7: Estimated Log-Likelihood t=1 to t=25

1.3 Estimation of Robot's past positions

We used smoothing technique to estimate the robot's past positions. In this technique we use the present sensor observations as well to estimate the past positions, thus giving us a better estimate. This can also be observed in the heat map in Figure 1.8 that the estimates after smoothing are less spread for even the initial positions of the robot. Smoothed and Ground Truth Locations are plotted in Figure 1.9

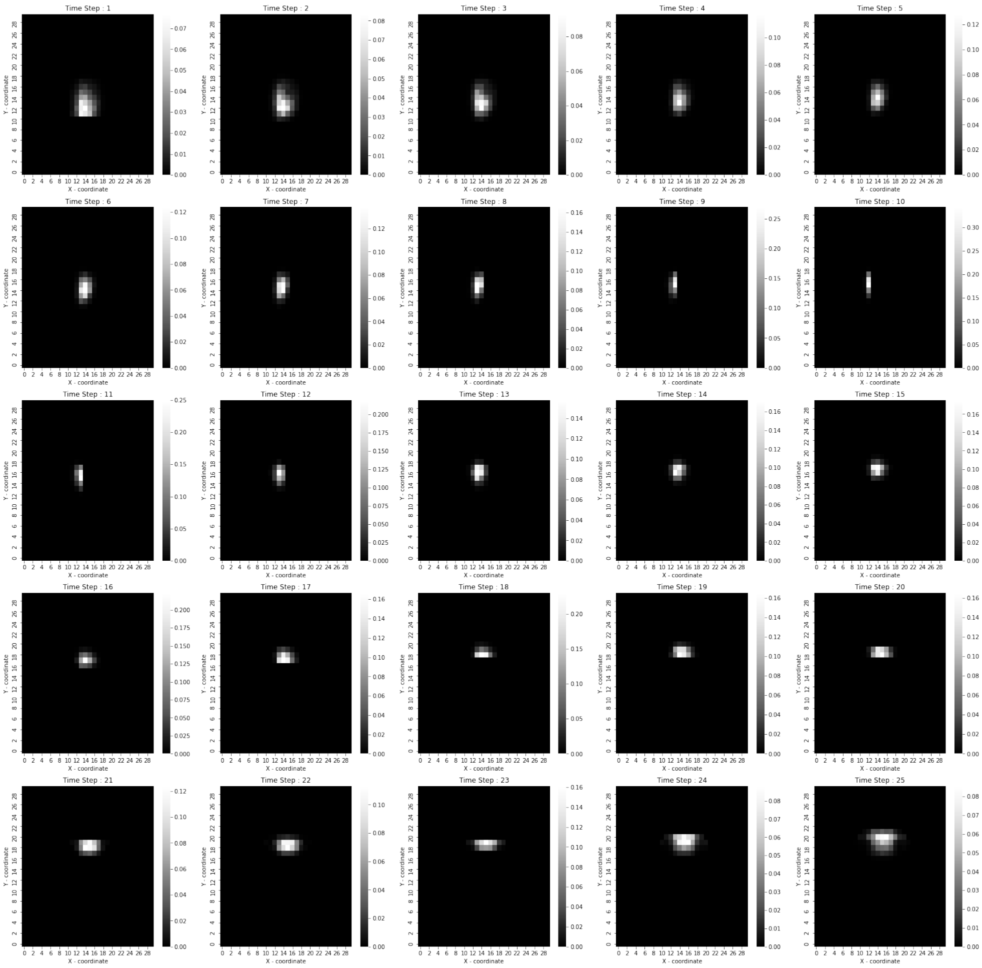

Figure 1.8: Smoothing Heat Map t=1 to t=25

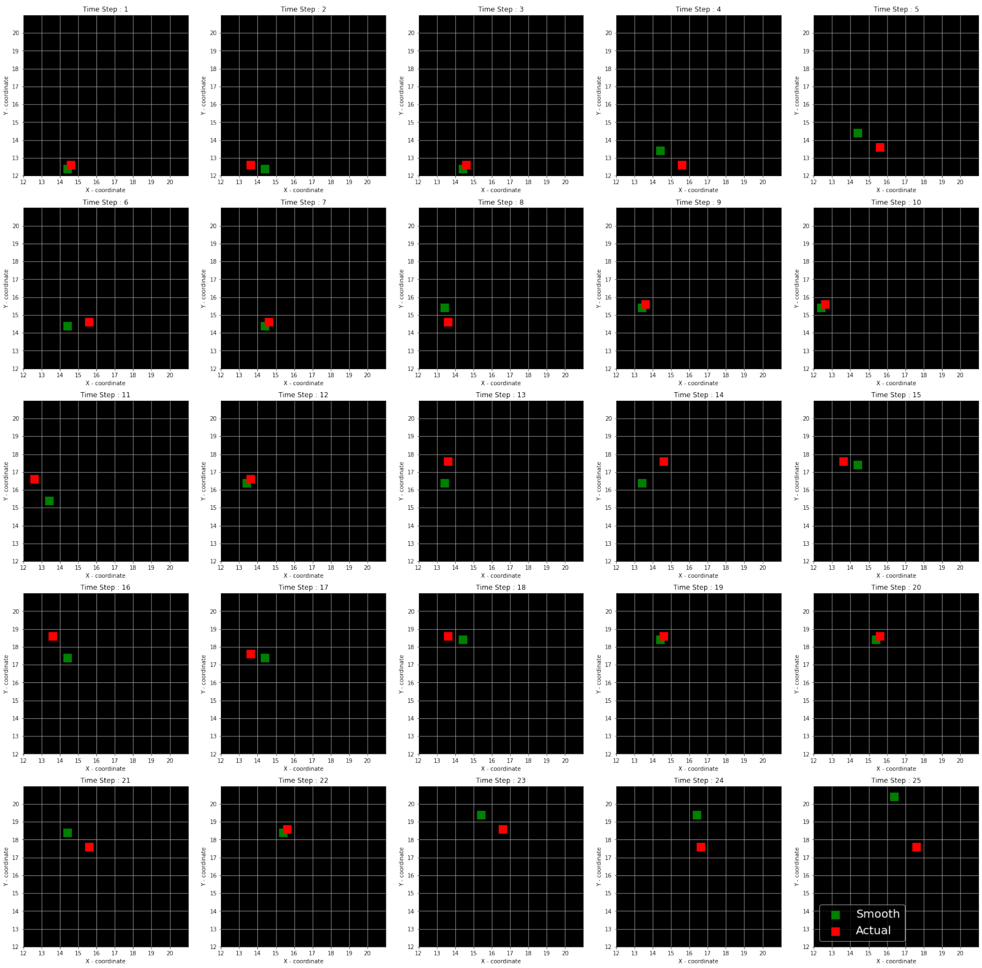

Figure 1.9: Smoothed and Ground Truth Locations: t=1 to t=25

1.4 Manhattan Distance between Actual and Estimated path

Total Manhattan Distance between the Estimated and Actual path is 67. In Figure 1.10, we can see that the error is on the higher side in the initial positions of the robot and decreases afterwards. The reason is that initially we don't have many sensor observations, which leads to more uncertainty and thus high values of Manhattan Distance are observed initially.

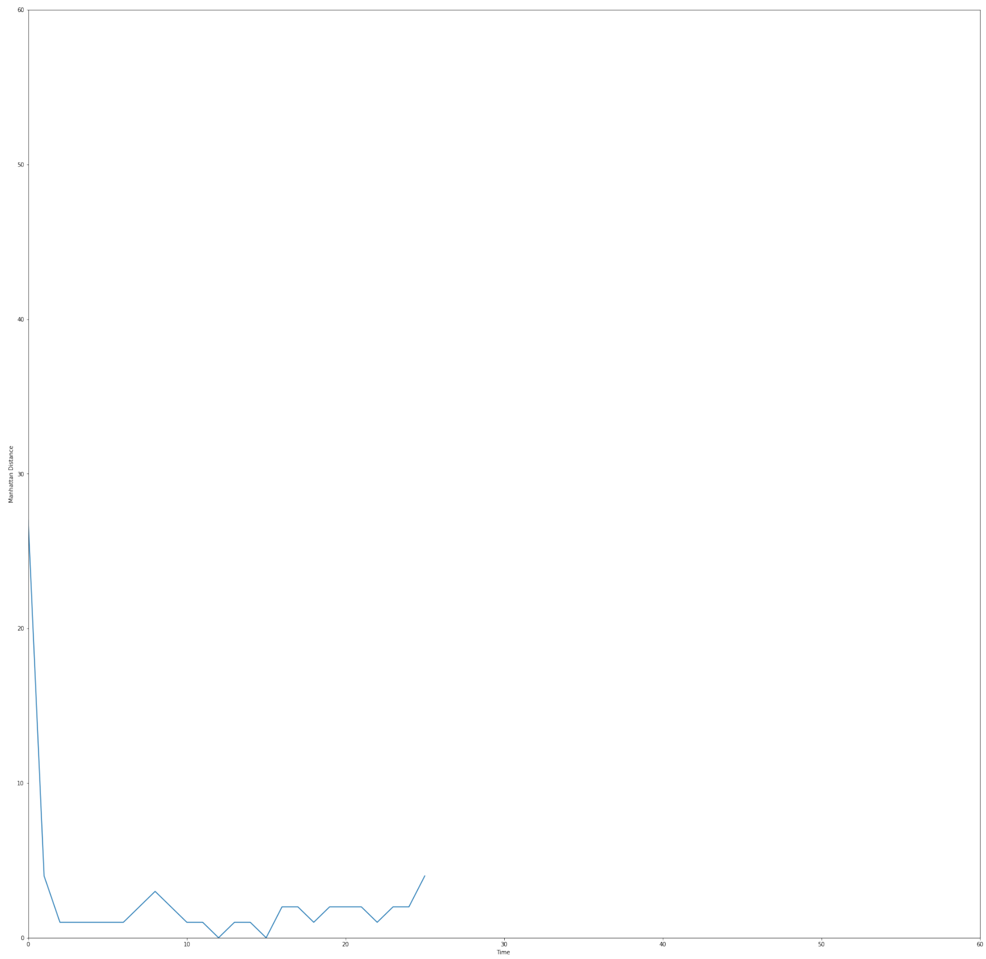

Figure 1.10: Manhattan Distance between Filtered and Actual Path

Total Manhattan Distance between the Smoothed and Actual path is 29. In Figure 1.11, we can see that the error significantly reduces in the initial positions of the robot as now we are using the present sensor observations to estimate the past positions.

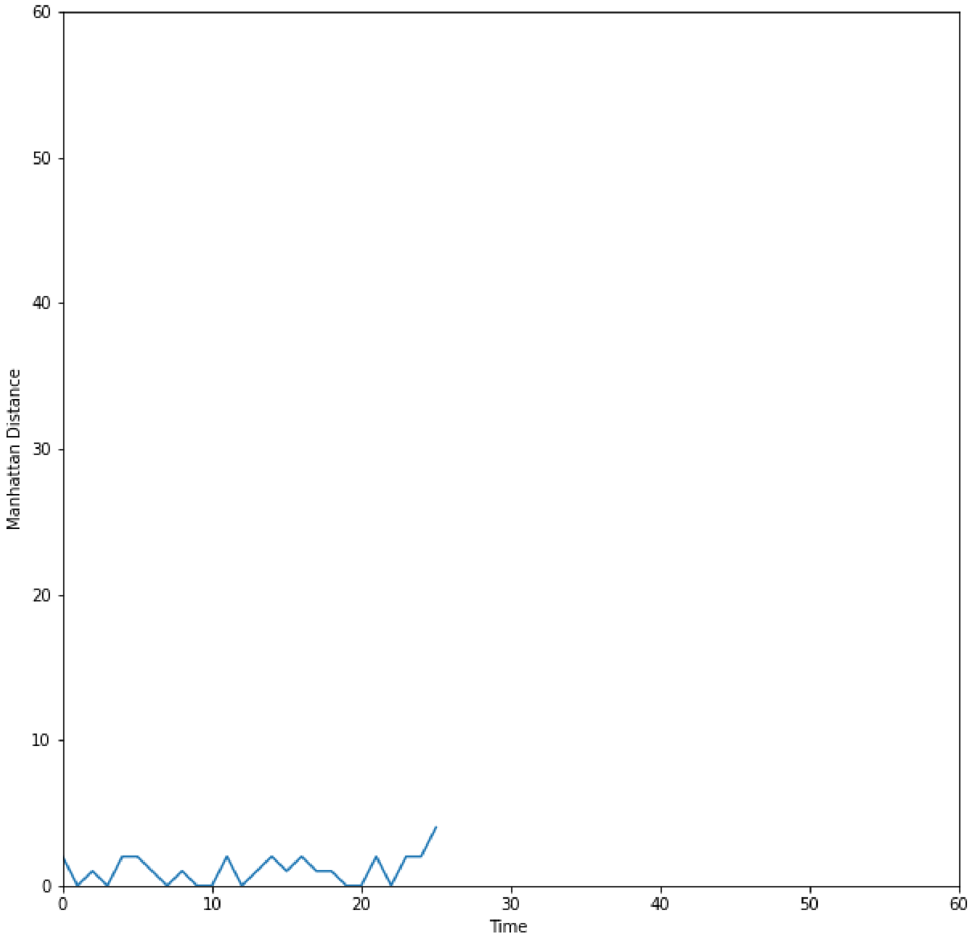

Figure 1.11: Manhattan Distance between Smoothed and Actual Path

1.5 Prediction of Robot's future location

From the predictive likelihood plots in Figure 1.12 and Figure 1.13, we observe that the position of the robot becomes more uncertain with time. The reason is that we don't have the sensor observations during this time period and with time this effect propagates and makes the likelihood more uncertain with time.

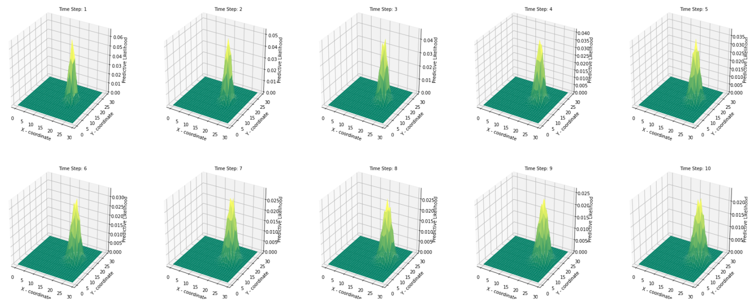

Figure 1.12: Predictive Likelihood over 10 time steps

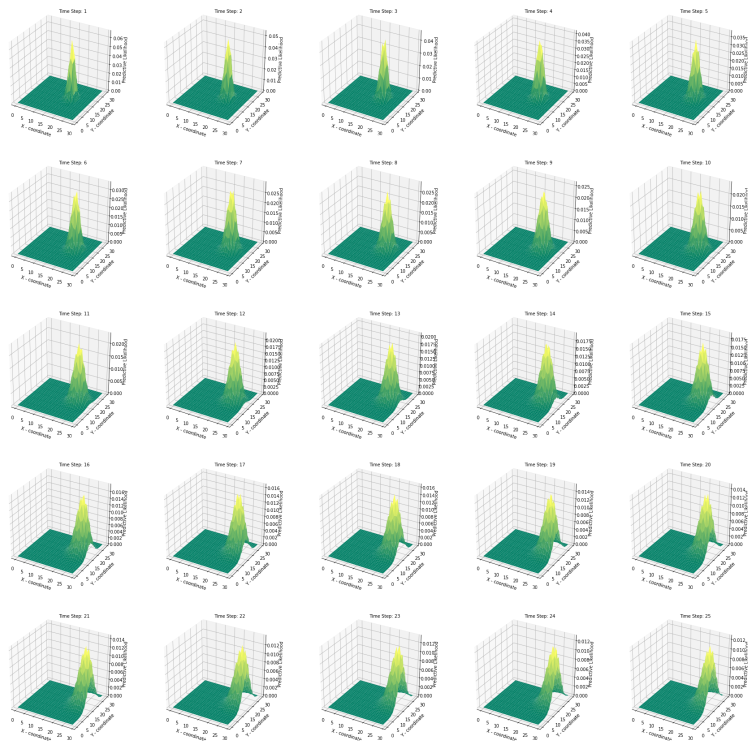

Figure 1.13: Predictive Likelihood over 25 time steps

1.6 Most Likely Path followed by the Robot

The Most Likely Path obtained using Viterbi algorithm can be seen in Figure 1.14. The Manhattan Distance between the Actual Path and the Most Likely Path comes out to be 70. The plot for the same can be seen in Figure 1.15. The Manhattan Distance is more than the one obtained between Smoothed and Actual Path because here we have one more constraint i.e. the locations should be adjacent to each other at t=i and t=i+1 for all t.

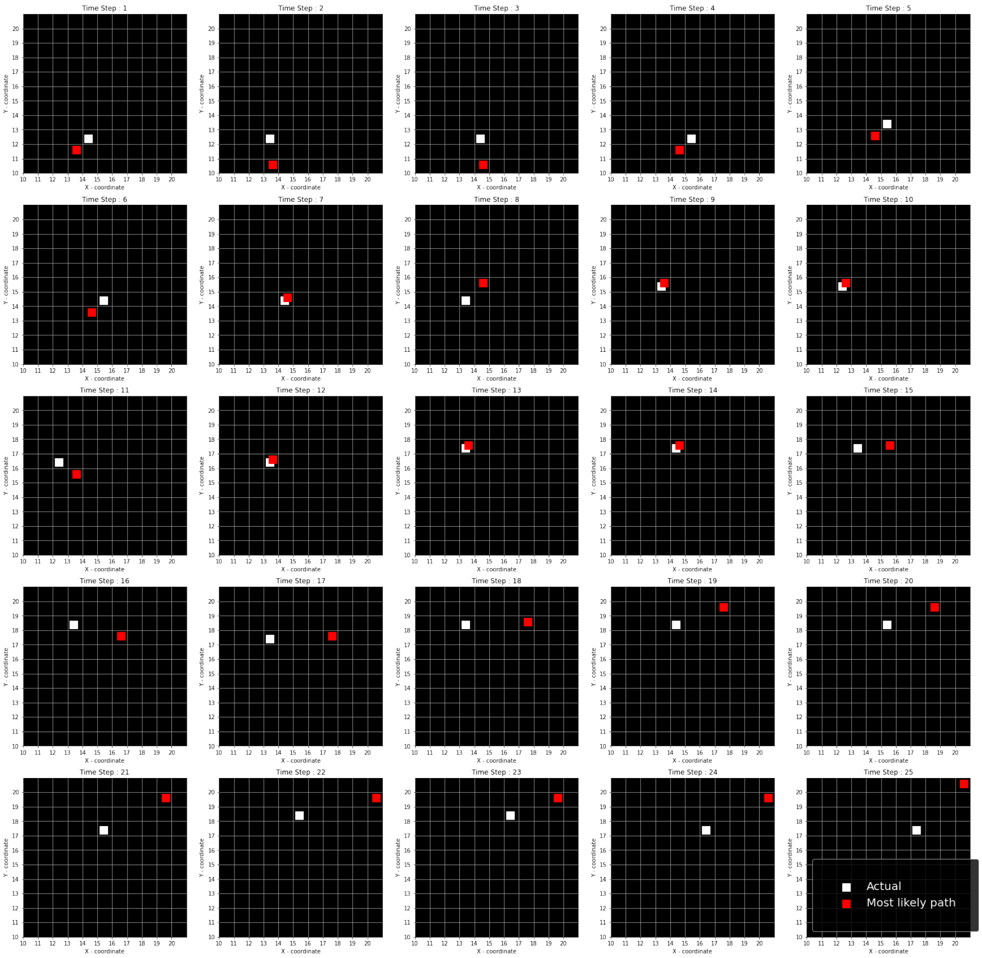

Figure 1.14: Most Likely path and Actual Path t=1 to t=25

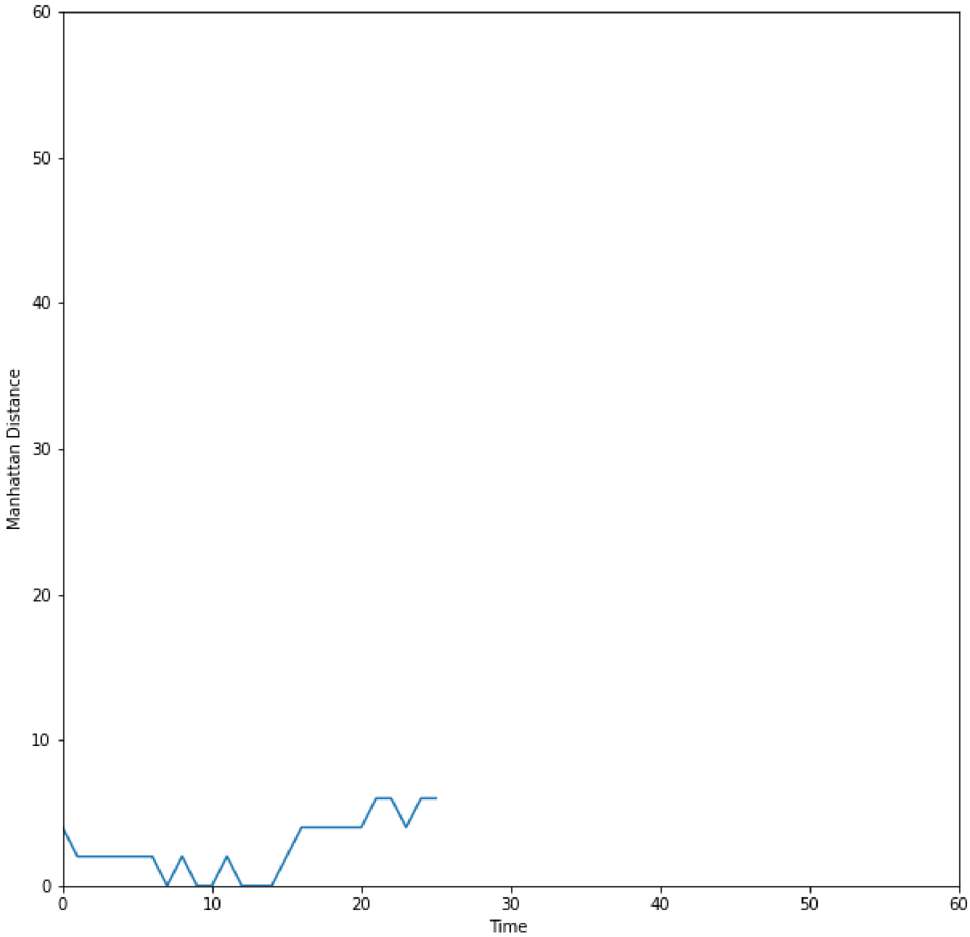

Figure 1.15: Manhattan Distance between Most Likely and Actual Path

Chapter 2

Continuous State Space Problem

The initial starting point of the agent is (60,20). The initial velocity is (2,2). The initial belief of position is at (10,10). And initial velocity belief is (1,1).

2.1 Motion model & sensor model

We construct motion model and sensor model as required. The actual and observed trajectory can be seen in 2.1 To get a better understanding, of what the values are at a particular

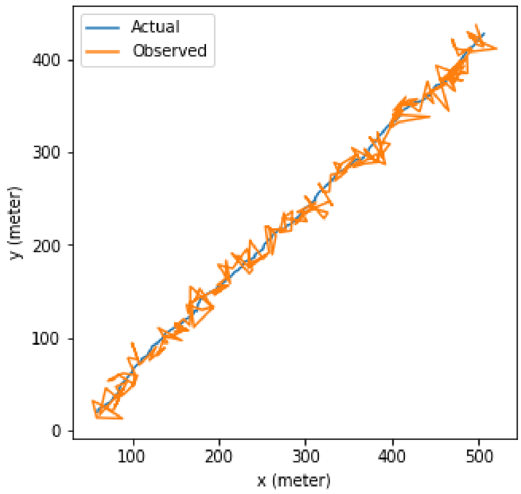

Figure 2.1: Actual & Observed Trajectory

timestep, we plot Figure 2.2. Here the points are plotted 20 time steps apart. i.e. t = 0, t = 20 so on.

2.2 Implement Bayes Filter

We implement the Bayesian Filter, with the specifications mentioned in the question. The plots obtained are in the next section

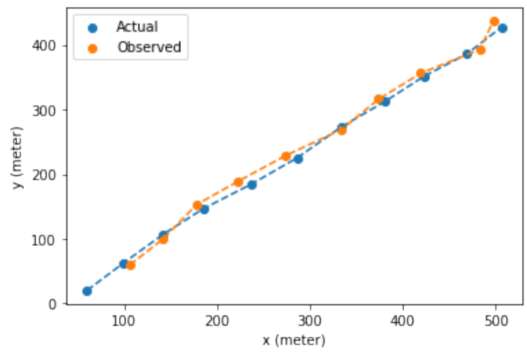

Figure 2.2: Actual & Observed Trajectory separated by 20 timesteps

2.3 Actual trajectory & estimated trajectory

Figure 2.3, shows the actual and estimated trajectory. The estimated trajectory is initially far away, but soon catches with the actual trajectory, as it obtains more and more sensor information

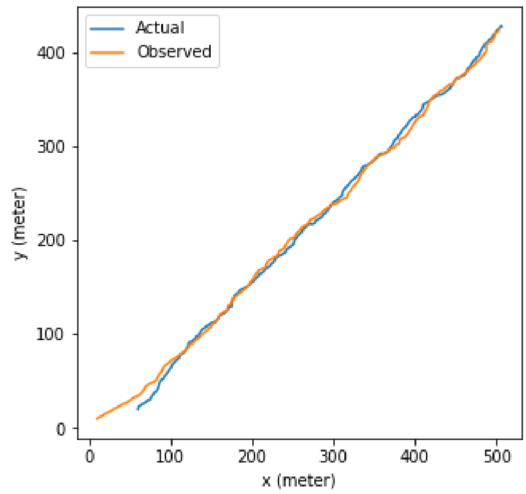

Figure 2.3: Actual & Estimated Trajectory

To get a better understanding of the exact values at particular time instants, We plot actual and estimated trajectory in Figure 2.4. Here the consecutive readings are 20 time steps apart. You can see, at t = 200 the mean of the belief distribution, is very close to the actual state.

Uncertainty ellipses are drawn at the gap of 20 timesteps in the Figure 2.5. You can see that the final uncertainty ellipse at t=200, contains the actual trajectory point, unlike initial timesteps. Hence the belief gets better with time.

The Figure 2.6, shows the actual, observed and the estimated trajectory.

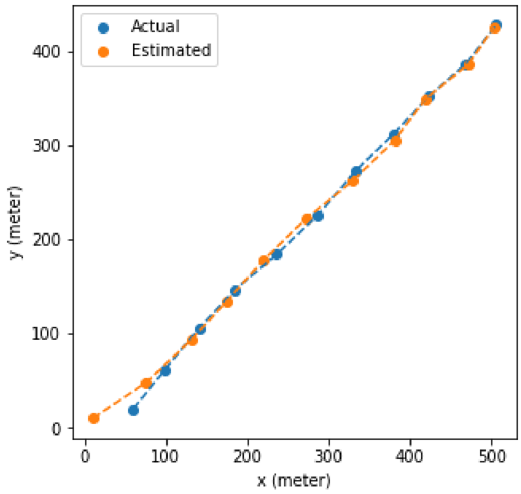

Figure 2.4: Actual & Estimated Trajectory separated by 20 timesteps

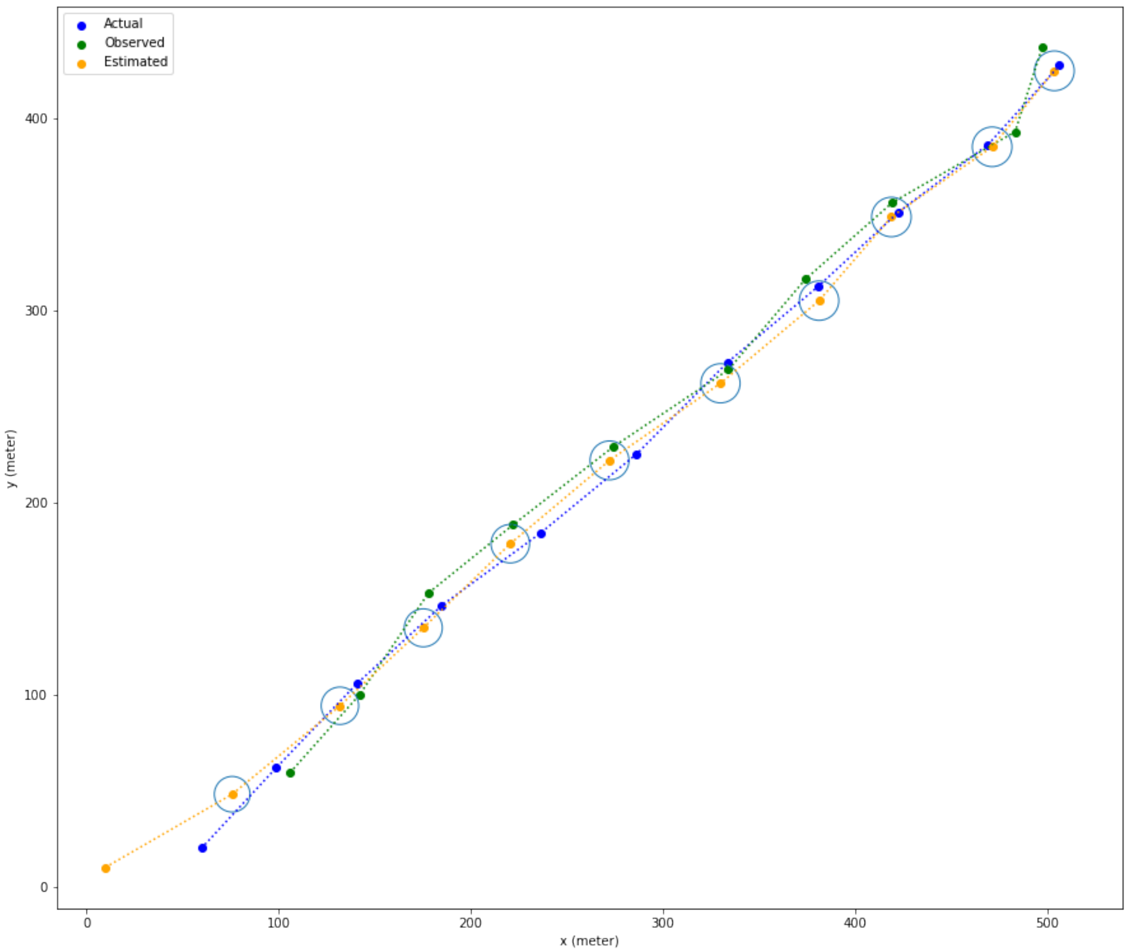

Figure 2.5: Actual, Observed & Estimated Trajectory with uncertainty ellipses

The Figure 2.7, shows the distribution of the final belief, at t = 200.

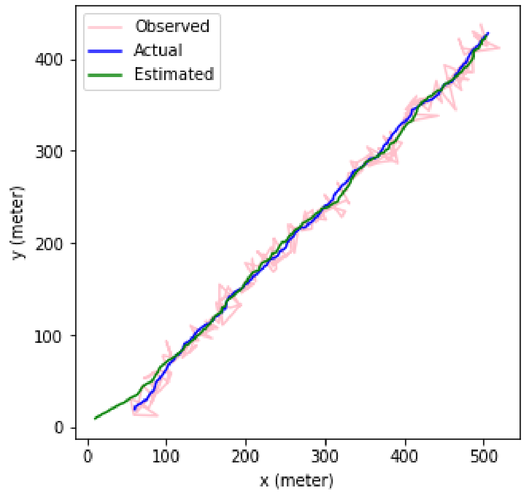

Figure 2.6: Actual, Observed & Estimated Trajectory

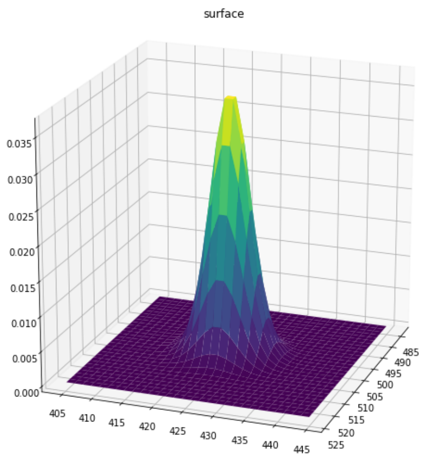

Figure 2.7: Belief at t = 200

2.4 Sinusoidal Control Policy

In this part we implement sinusoidal control policy. The control velocity i.e. v_x is $sin(\theta)$ and v_y is $cos(\theta)$. The value of θ at time step t is $\theta = 30° * t$.

The continuous plots of actual,observed and estimated can be seen in the Figure 2.8. You can see the effect of sinusoidal control policy on the actual trajectory plot. It has small peaks and valleys.

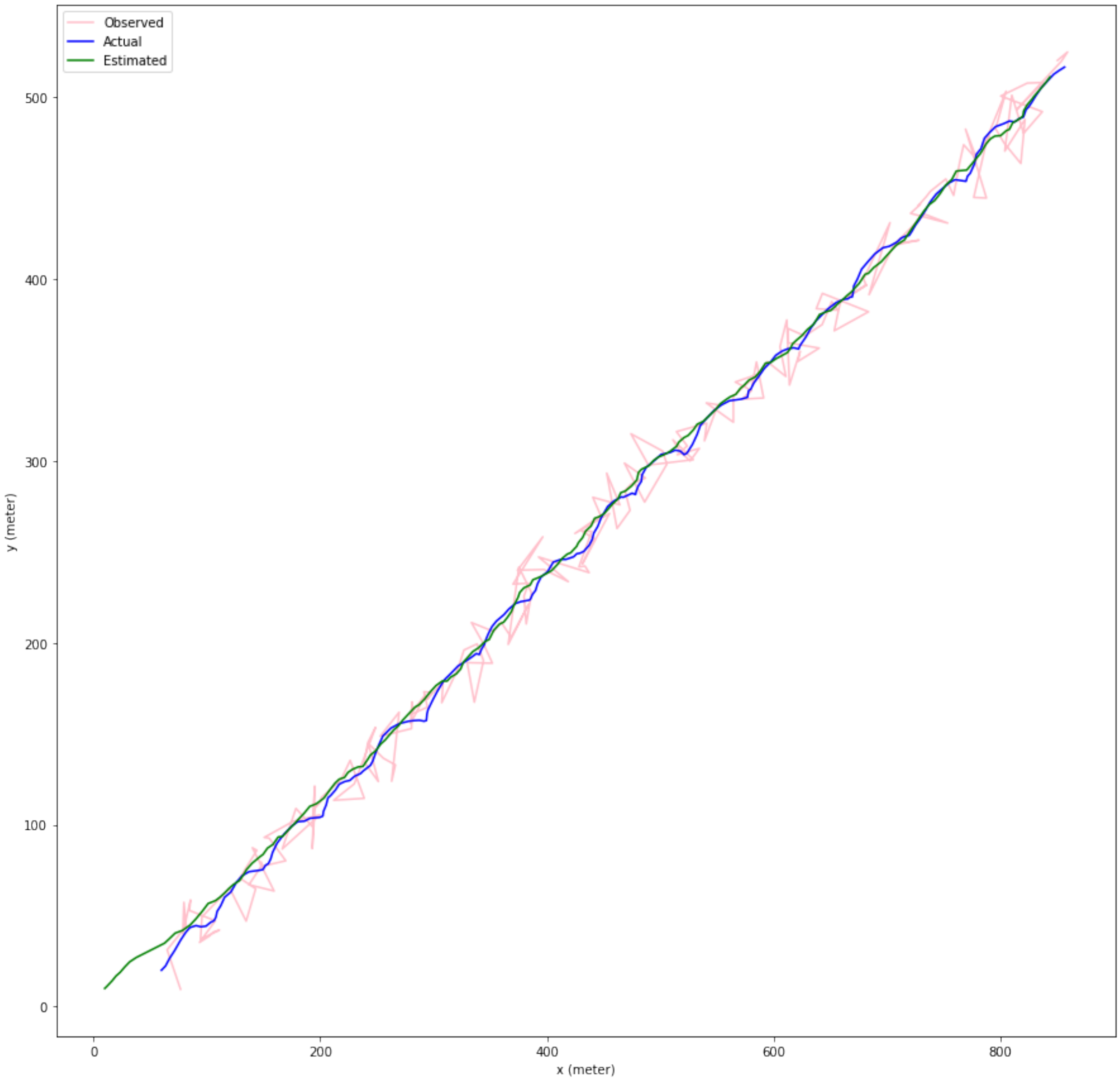

Figure 2.8: Actual, Observed & Estimated Trajectory

The plots of actual,observed and estimated can be seen in the Figure 2.9. Here the observations are plotted, at the gap of 20 time steps i.t. t = 0, t = 20 so on. You can also see the uncertainty ellipses here. The final uncertainty ellipse, also contains the actual trajectory point.

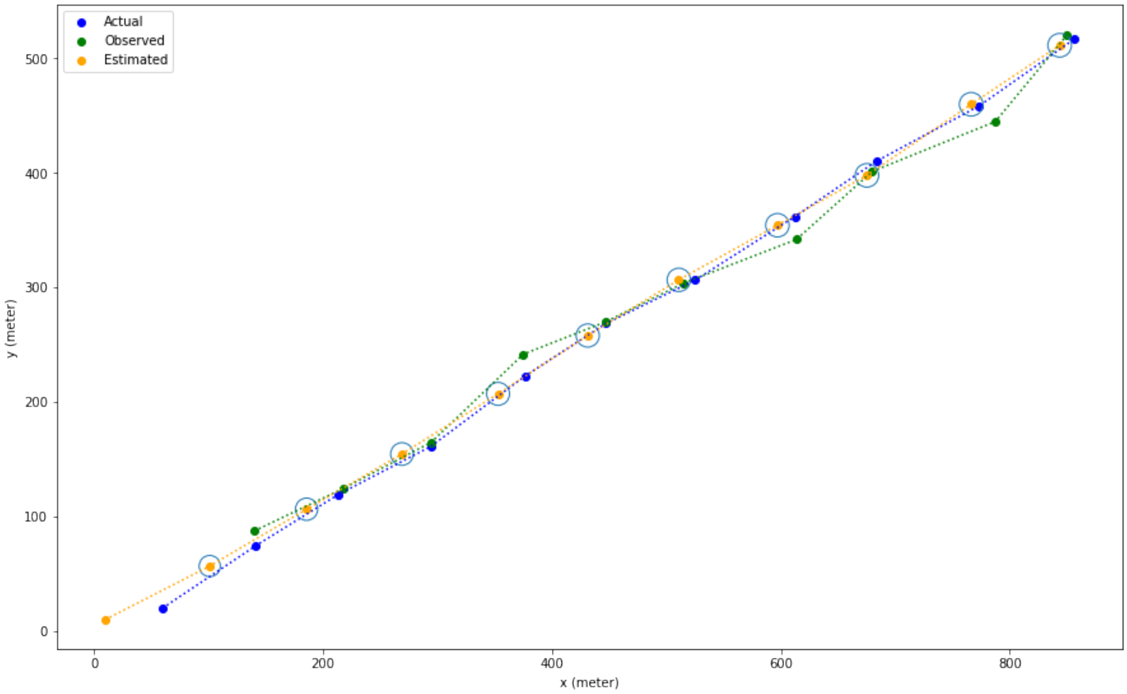

Figure 2.9: Actual, Observed & Estimated Trajectory with uncertainty ellipses

The euclidean error plot, can be seen in the Figure 2.10.

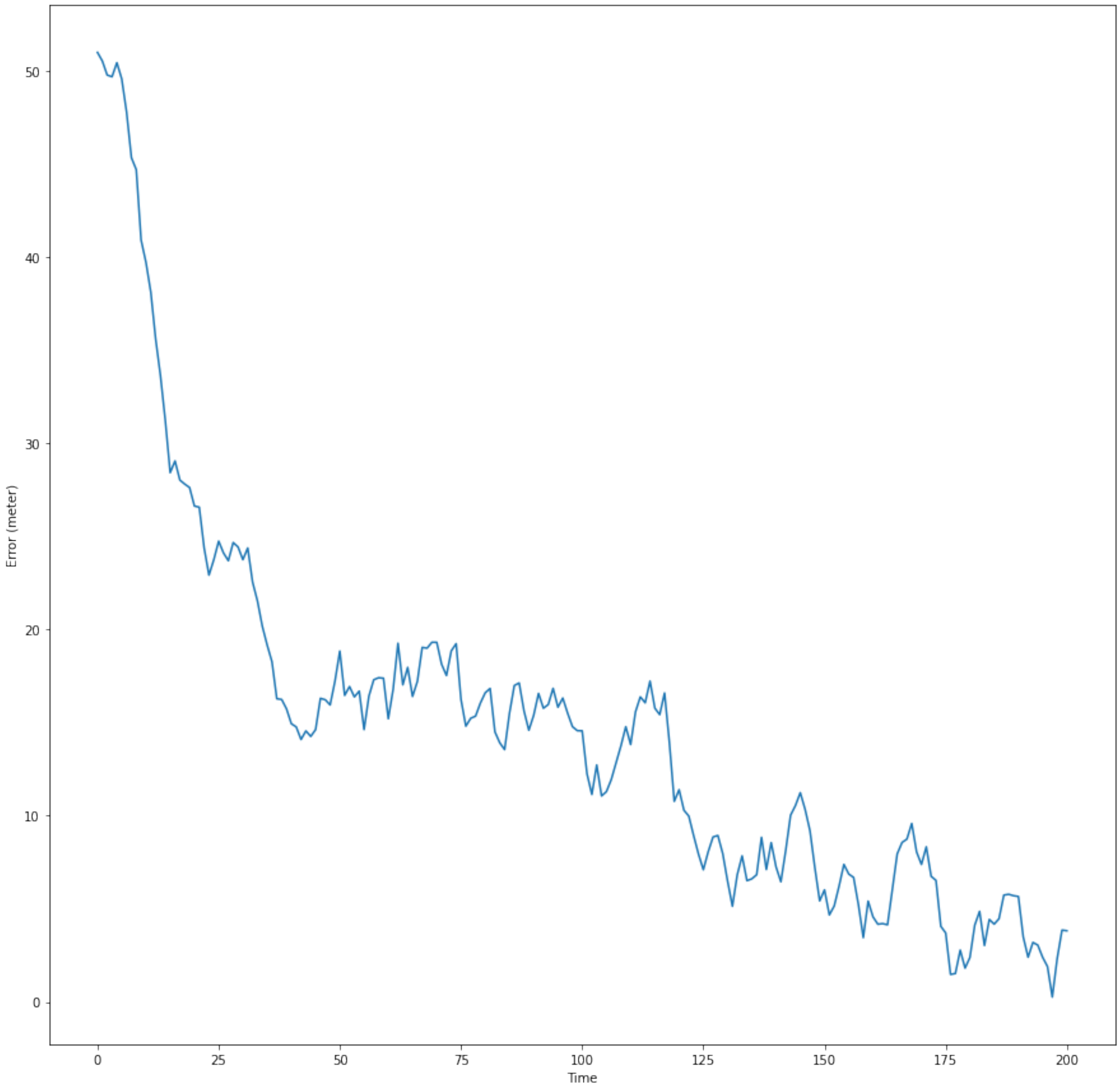

Figure 2.10: Euclidean Error plot

2.5 Effect of varying uncertainty of sensor

In this we vary the uncertainty of sensor model. Let β be the diagonal element of the noise covariance matrix (R) for the sensor model. Originally the value was $\beta = 100$.

- $\beta = 10000$

  We get the estimated trajectory plot as in the Figure 2.11.

  Euclidean error plot can be seen in the Figure 2.12.

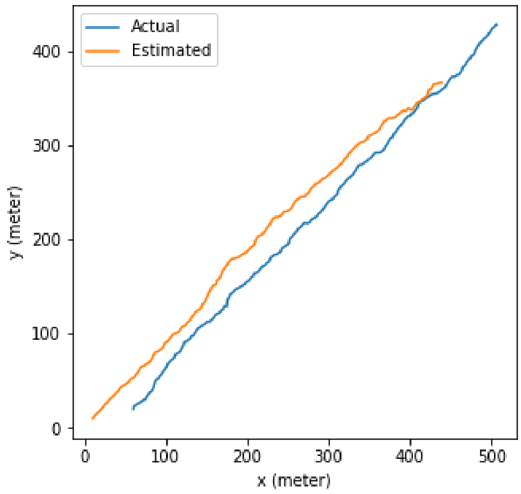

Figure 2.11: Actual & Estimated Trajectory

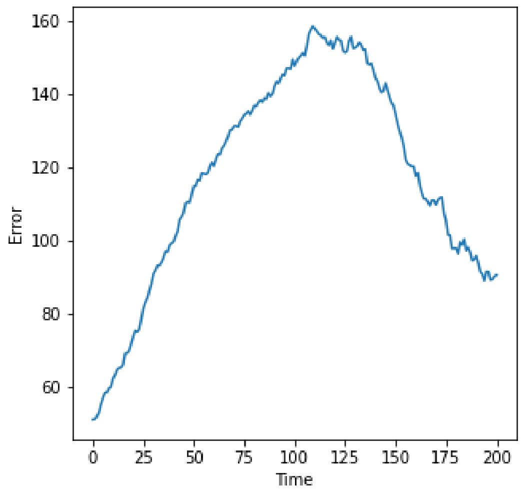

Figure 2.12: Euclidean Error

- $\beta = 1000$

  We get the estimated trajectory plot as in the Figure 2.13.

  Euclidean error plot can be seen in the Figure 2.14.

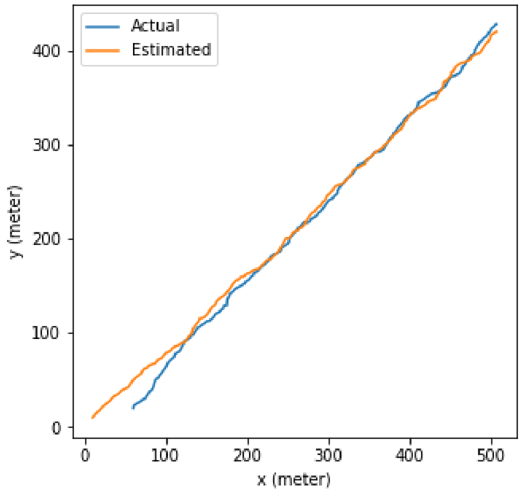

Figure 2.13: Actual & Estimated Trajectory

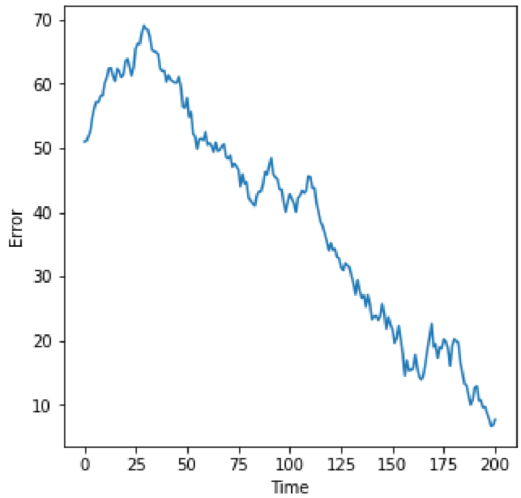

Figure 2.14: Euclidean Error

- $\beta = 10$

  We get the estimated trajectory plot as in the Figure 2.15.

  Euclidean error plot can be seen in the Figure 2.16.

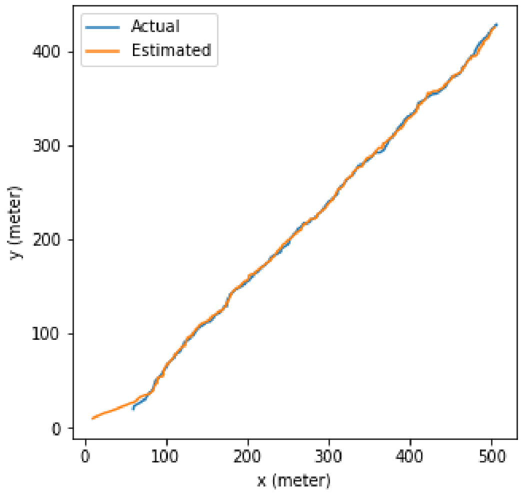

Figure 2.15: Actual & Estimated Trajectory

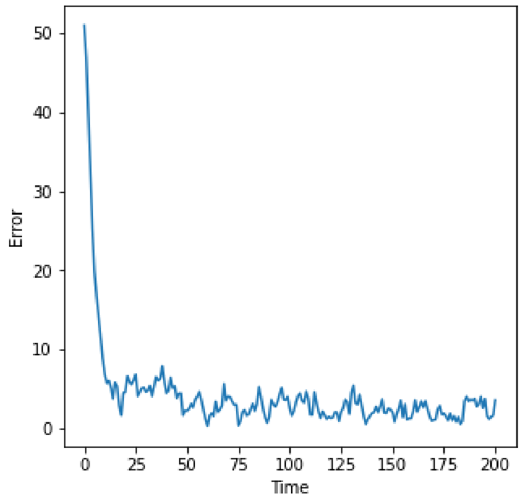

Figure 2.16: Euclidean Error

- $\beta = 1$

  We get the estimated trajectory plot as in the Figure 2.17.

  Euclidean error plot can be seen in the Figure 2.18.

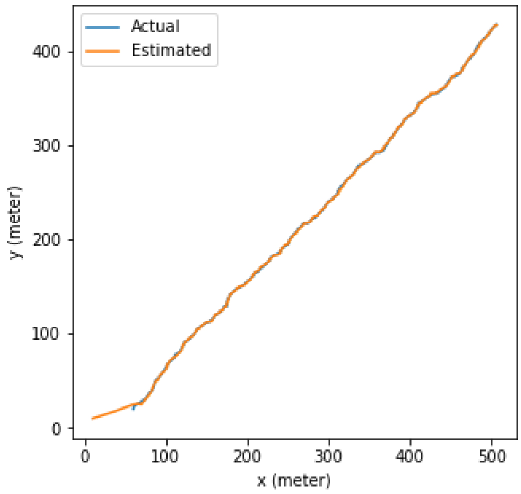

Figure 2.17: Actual & Estimated Trajectory

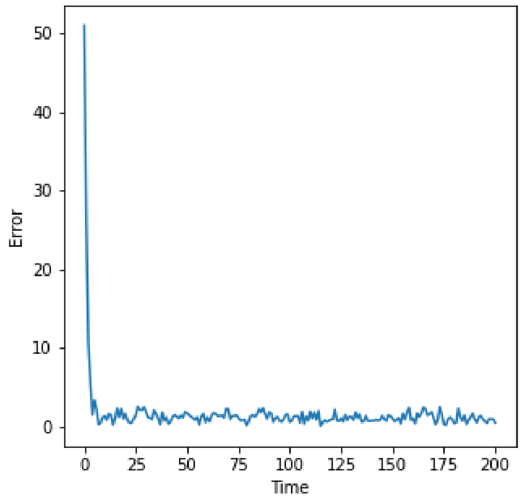

Figure 2.18: Euclidean Error

From the above plots, we can see that as the value of β is decreased, the estimater gets better. It is also evident from the euclidean error plots.

2.6 Higher uncertainity over intial belief

We increase the uncertainty in position of initial belief to standard deviation of 100. We get the actual,observed and estimated curves as shown in the Figure 2.19. The higher uncertainty over initial belief, let's the belief, track faster, the actual trajectory. This behavior is also expected, because higher uncertainty implies, week prior, and sensor update will dominate the measurement update.

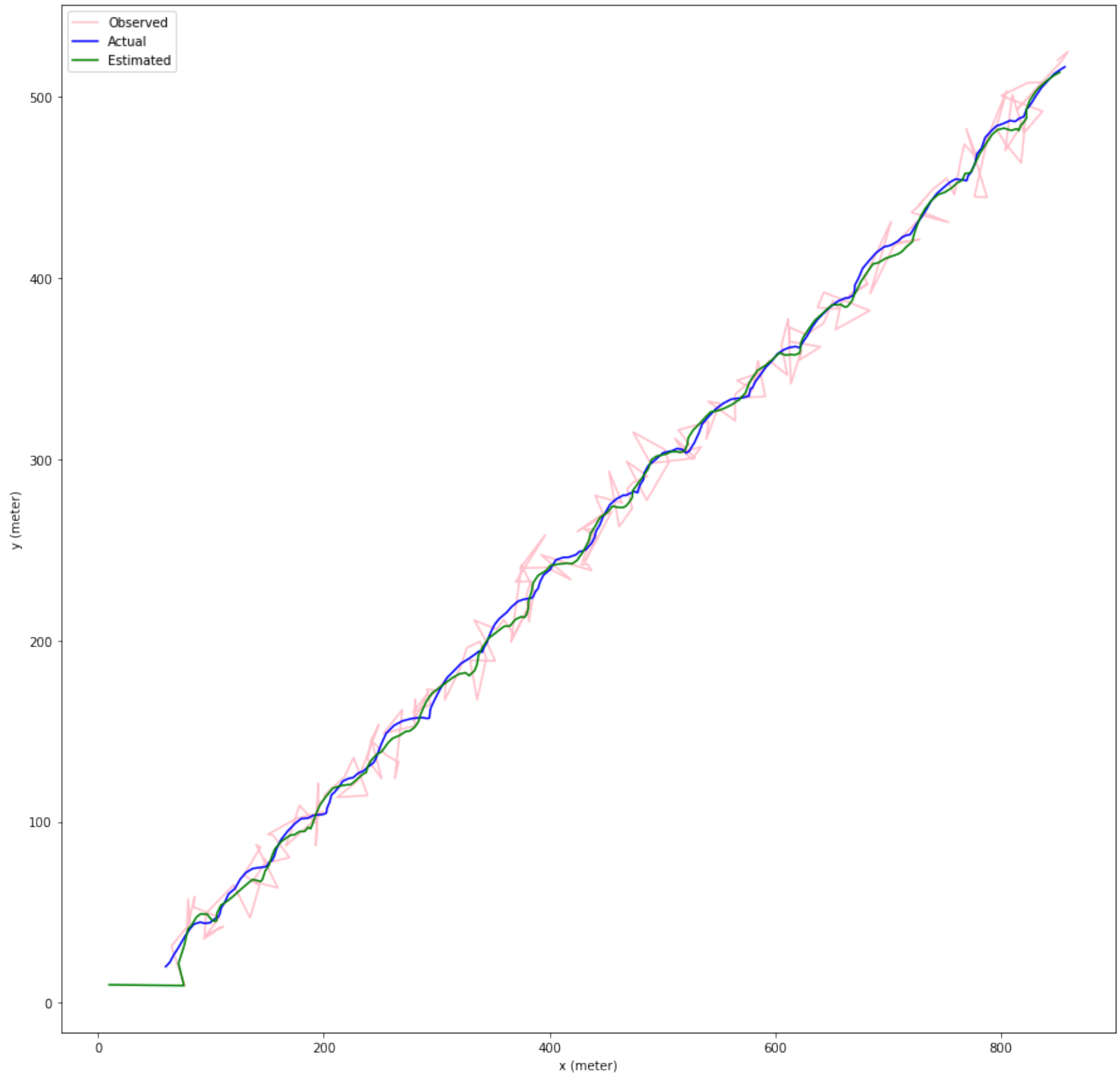

Figure 2.19: Actual, Observed and Estimated Trajectory with higher initial uncertainty

Figure 2.20 shows the euclidean error plot. You can see the error falls to very low value, initially. This is because, the sensor measurement dominates. This behavior is also expected.

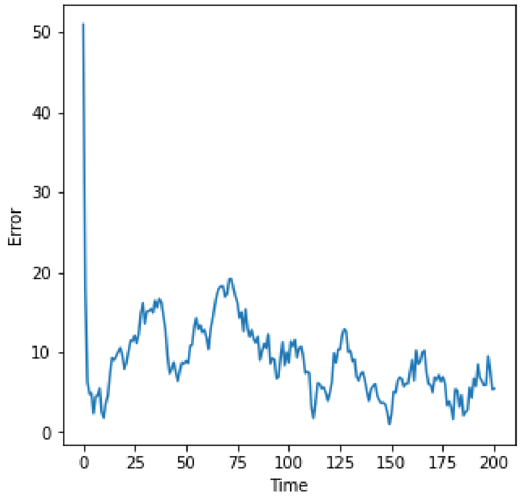

Figure 2.20: Euclidean Error

Figure 2.21 is for the original belief, where the uncertainty in the initial belief was low. You can see it takes more time initially for the belief to track unlike Figure 2.19. This is because of the string prior, initially in case of Figure 2.19.

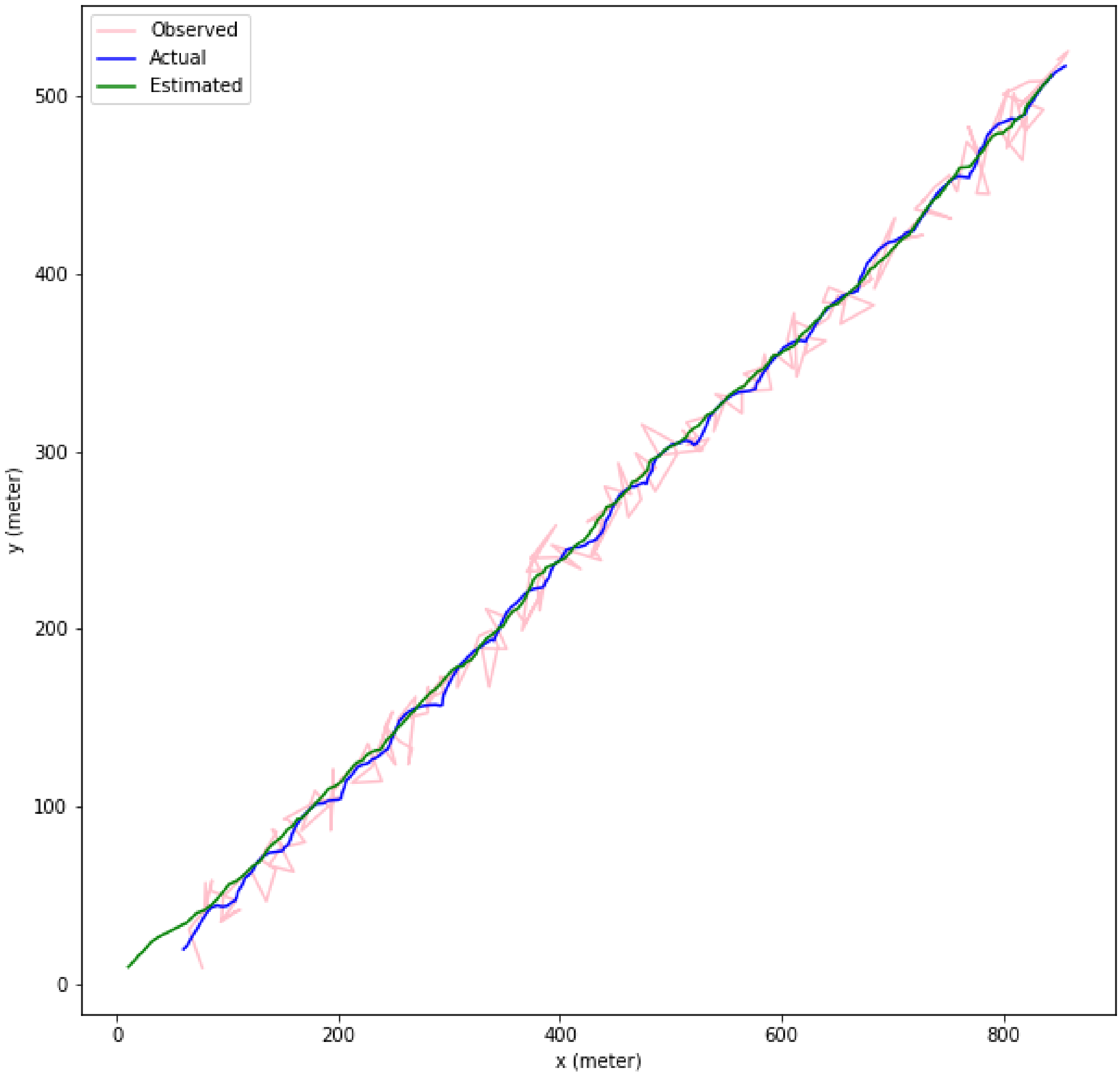

Figure 2.21: Actual, Observed and Estimated Trajectory

Figure 2.22 is the euclidean plot for our original initial belief(which had a strong prior/ decreased uncertainty). We can that error in the initial stages is quite high as compared to our new initial belief (which has a weak prior / increased uncertainty)

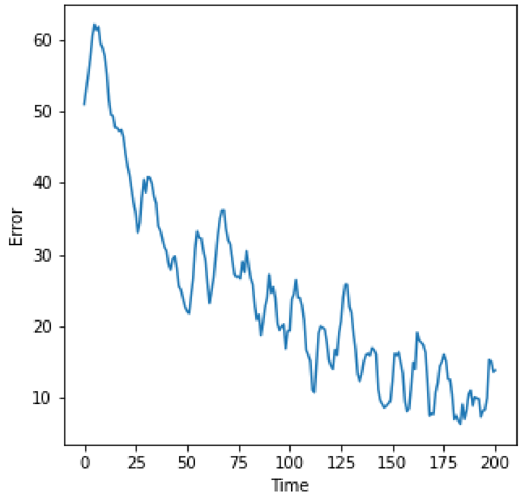

Figure 2.22: Euclidean Error

2.7 Drop out sensor observations

The sensor observations are dropped out at t = 10 and t = 30 for a period of time steps. In the region 10<=t<20 and 30<=t<40, we see that euclidean error increases as in the Figure 2.23. This behaviour is expected, because sensor observations, are used for correction to our belief. However, when they are absent, there is only motion model. This leads to increased error.

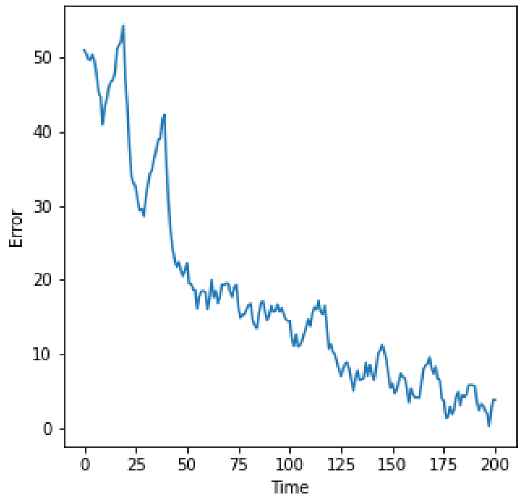

Figure 2.23: Euclidean Error

Uncertainty ellipses can be seen in the Figure 2.24. The size of ellipse increases for 10<=t<20, then decreases for 20<=t<30, then again increases for 30<=t<40, then decreases for 40<=t<50. The behavior is expected, because, sensor information incorporation, also has the effect of reducing uncertainty. Clearly when 10<=t<20 and 30<=t<40, when the sensor observations are absent, the uncertainty grows.

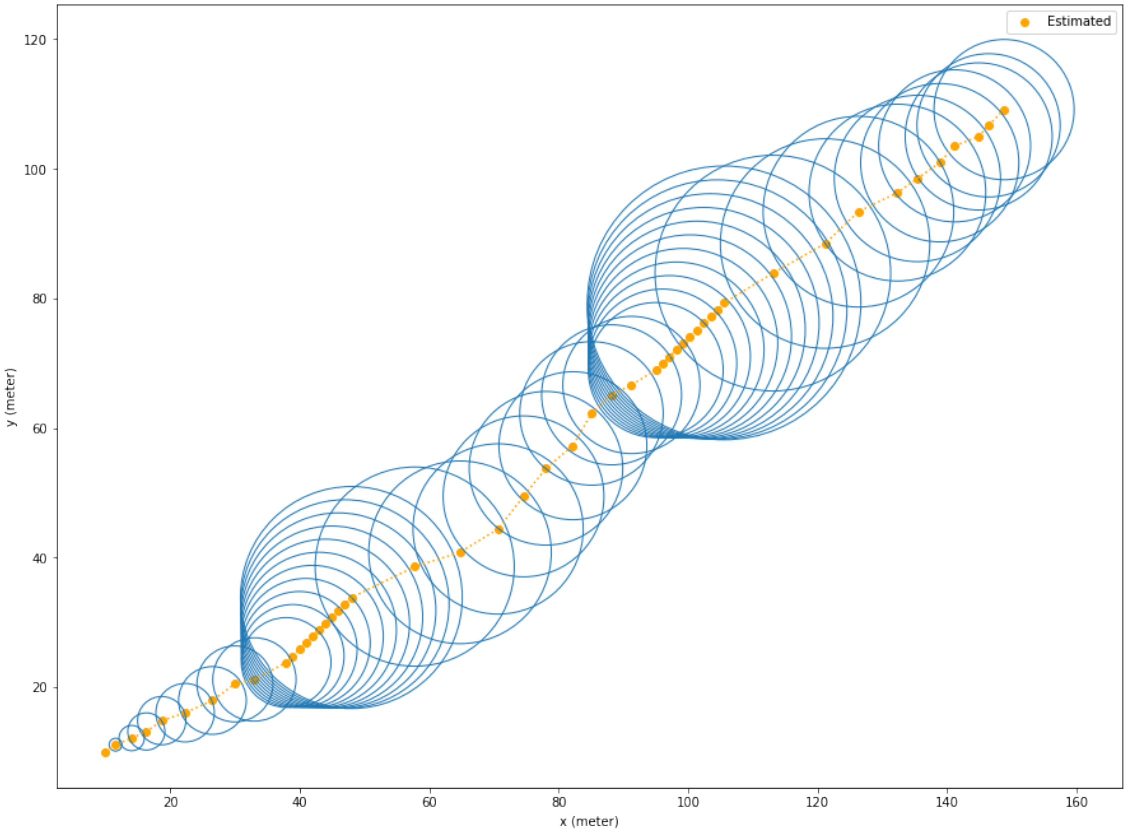

Figure 2.24: Uncertainty Ellipses for the Estimated trajectory

2.8 Estimated velocities and true velocities

We can see the estimated velocity and actual velocity plot in the Figure 2.25. The initial velocity estimate was (1 m/s, 1 m/s), whereas the actual initial velocity was (2 m/s, 2m/s). The plot in the Figure 2.25 is quite interesting, as the estimated curve is approaching the actual curve, as time passes by. Hence we can say that the estimator is able to track the velocity.

As we increased the timesteps from 200 to 400. The estimator was able to reach exactly 2 m/s.

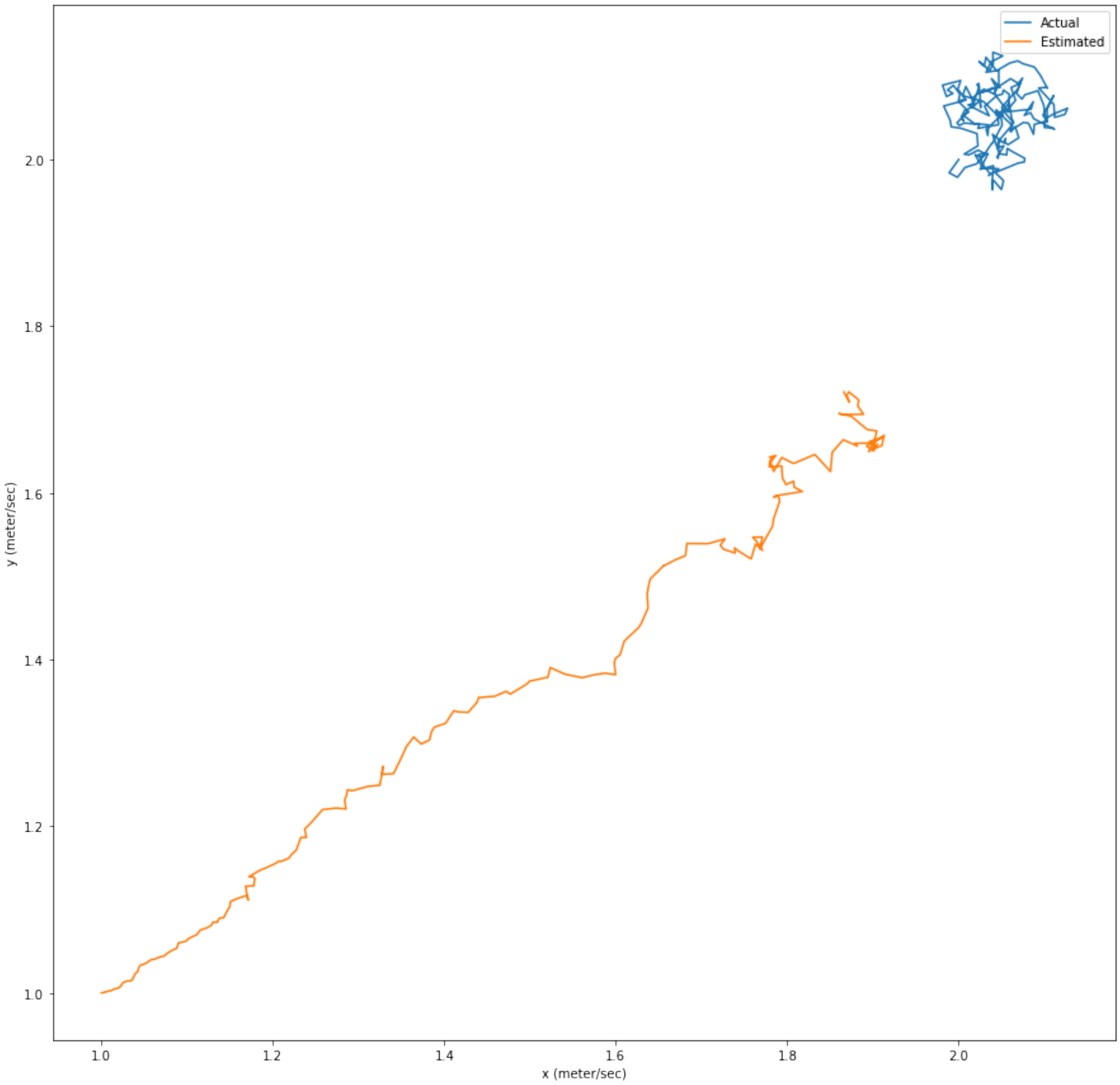

Figure 2.25: Actual and Estimated velocity plots

Figure 2.26 is the euclidean error metric between the velocities. As you can see, the error decreases, as time step increases. Hence we can safely say that the estimator can track the velocity. This part has no control policy.

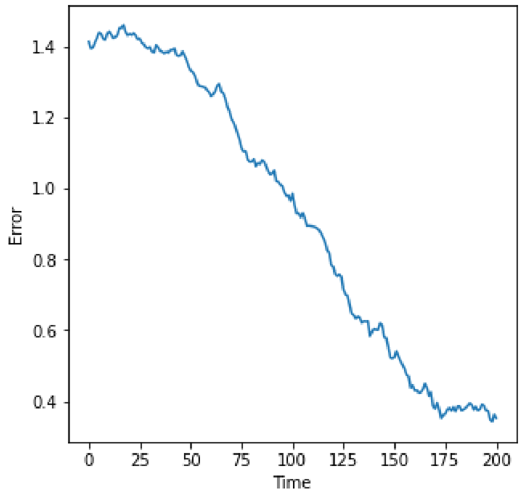

Figure 2.26: Euclidean Error

2.9 Data Association

For this part we use, **Global Nearest Neighbour** filter along with Kalman filter. At a particular timestep, we do the sensor observations assignment. This assignment is done such that after the assignment the sum of **Mahalanobis distances** from the predicted trajectories is minimum. This part has no control policy.

Starting Location of agent1 $= (60, 20)$

Starting Location of agent2 $= (20, 60)$

Estimated start location of agent1 $= (20, 10)$

Estimated start location of agent2 $= (20, 10)$

The noise model for agent to are kept twice the value of the agent 1. i.e. R2 = 2*R1 and Q2 = 2*Q1. We generate the two trajectories for the respective agents, as in Figure 2.27.

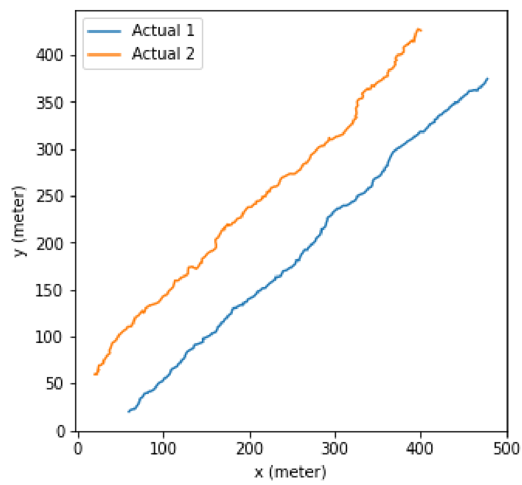

Figure 2.27: Actual Trajectories of both the agents

We get two values of sensor observations at each time step. We don't keep track of which observation is for which agent. The observations can be seen in Figure 2.28.

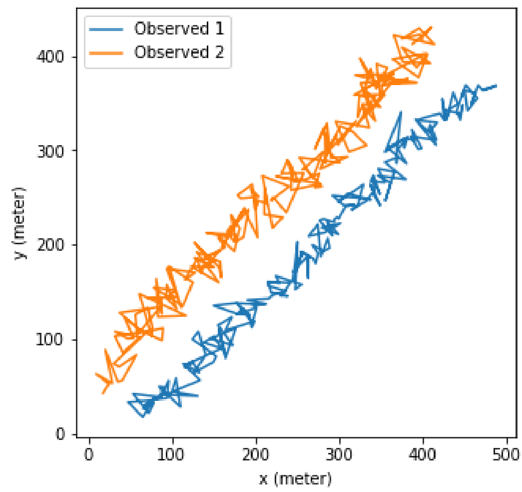

Figure 2.28: Observed Trajectories of both the agents

Now we apply, Kalman Filtering along with Global Nearest Neighbour Filter (for sensor observation assignment). We can see the estimated and actual paths in the Figure 2.29. As you can see, that our data associatin strategy works, and out estimated trajectory is able to track the actual trajectory for respective agents.

Our data association strategy i.e. **Global Nearest Neighbour** filtering is quite famous and effective. It can be generalized easily to 4-5 agents. All it does is at any particular instant, assigns the sensor measurements. In the case of 2 agents, it was pretty easy, because of two cases only, hence it was possible to check manually. To scale it to 4-5 agents, the number of possible combinations can go to 4! - 5!. To effectively do we can either use **Max-Flow Algorithm** or we can use **Hungarian Algorithm**

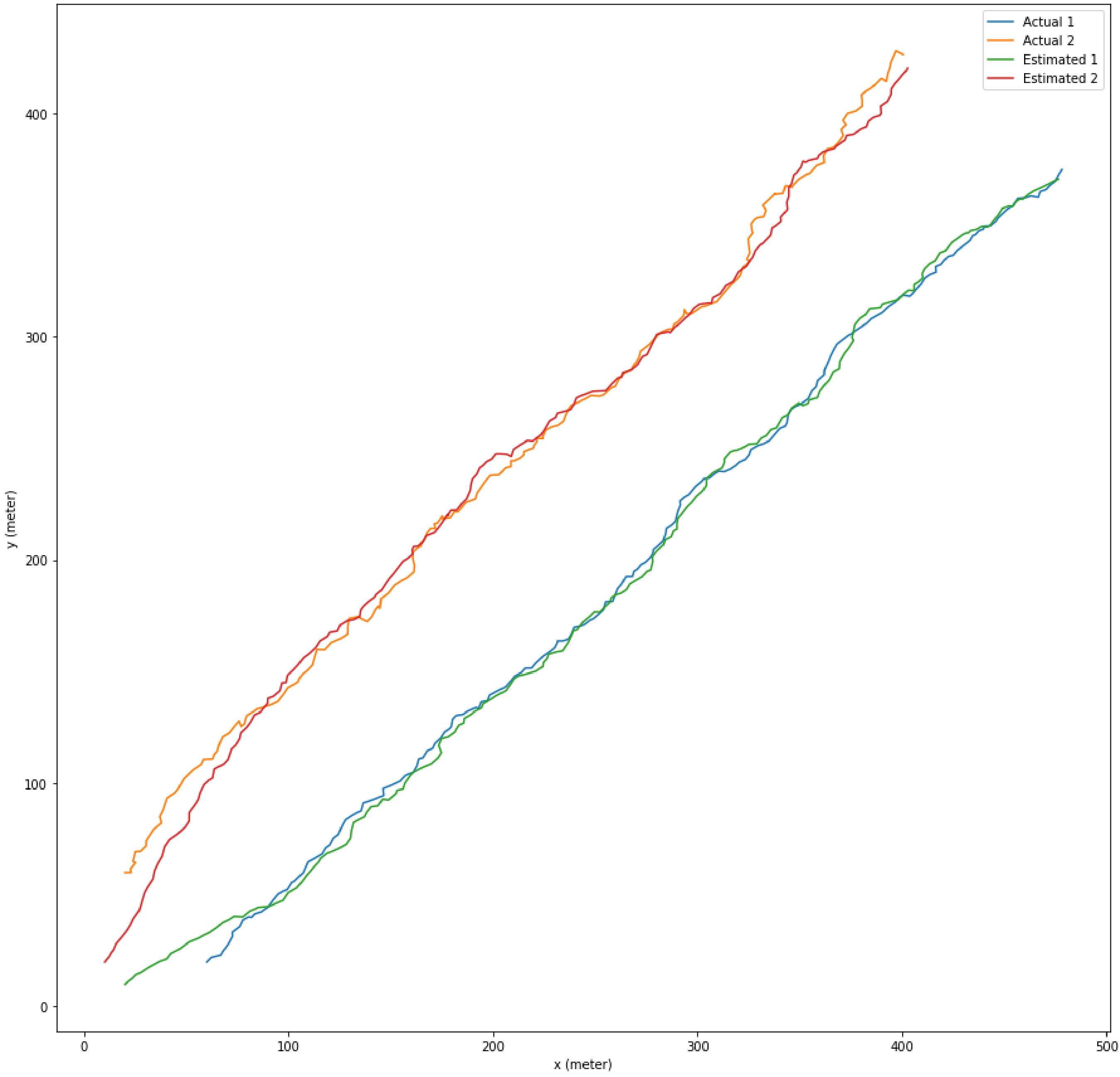

Figure 2.29: Actual & Estimated Trajectories of both the agents

Chapter 3

Conclusion

3.1 Discrete State Space Problem

In this problem, we applied four main operations on the actual path:
1) Filtering: Estimating the state, given all evidence to the present.
2) Smoothing: Estimating a past state, given all evidence up to the present
3) Prediction: Estimating a future state, given all evidence up to the present.
4) Most Likely Path: Sequence of states that is most likely to have generated those observations.
Various Heat Maps and 3D plots were obtained to for better analysis and visualisation. Results obtained were very intuitive and quite explainable. The individual conclusion to each of the subpart has been included in Chapter 1.

3.2 Continuous State Space Problem

We applied Kalman filter to continuous state space problem. We ran various experiments such as varying uncertainty of sensor model, uncertainty of motion model, sinusoidal control policy, dropping sensor observations, uncertainty ellipses, data association. The results obtained were quite intuitive, and have a logical explanation. The individual conclusion to each of the subpart has been included in the Chapter 2.

Bibliography

[1] Stuart J. Russell and Peter Norvig. 2003. Artificial Intelligence: A Modern Approach (3rd ed.). Pearson Education.

[2] S. Thrun, W. Burgard, and D. Fox, Probabilistic Robotics (Intelligent Robotics and Autonomous Agents), The MIT Press, Cambridge, MA, USA, 1st edition, 2005.