COL864 - PLANNING AND ESTIMATION FOR AUTONOMOUS SYSTEMS ASSIGNMENT 1

Name: RAVINDRA ABHAY KUDCHADKAR Entry Number: 2018TT10941

PART 1 : Hidden Markov Model for State Estimation including discrete variables

1. **State Space and Observation Space:**

   The state space for the vehicle is the discretized 2D grid (30 x 30). The state variable of the vehicle $\mathbf{X_t}$ can assume any value in the domain of the grid.

   The observation space has 4 observation variables, one from each of the 4 sensors. Each sensor tells with a particular probability, the presence of the vehicle in the grid cell.

2. **Transitional Model and Observational Model:**

   The transitional probability from the state $X_{t-1} = x_i$ to the state $X_t = x_j$ can be written as :

$$T_{i,j} = P(X_t = x_j | X_{t-1} = x_i)$$

   .

   $T_{i,j} = 0.4$ if x_j is above x_i. $T_{i,j} = 0.1$ if x_j is below x_i. $T_{i,j} = 0.2$ if x_j is to the left of x_i. $T_{i,j} = 0.4$ if x_j is to the right of x_i. This is because the vehicle can make 4 discrete actions of moving up, down, left, right.

   This gives a 900 x 900 transition matrix $\mathbf{T}$ which tells us the probability of transition from one state to another.

   The Observation probability (which helps us the infer about the position of the vehicle) for a particular observation (whether a particular sensor returns the presence of the object) $E_t = e$ given that the vehicle is in state $X_t = x_i$ is given by

$$O_{e,x_i} = P(E = e | X_t = x_i)$$

   . According to the given likelihood and the locations of the 4 sensors, the can be a total of 9 different types of observations. Let S_1 be the leftmost sensor, S_2 the centre, S_3 the rightmost and S_4 the topmost. Let s_i denote the True observation (presence) for sensor i and $\bar{s_i}$ be the False (absence). The possibilities are :

$$s_1 \cap \bar{s_2} \cap \bar{s_3} \cap \bar{s_4}$$

$$s_2 \cap \bar{s_1} \cap \bar{s_3} \cap \bar{s_4}$$

$$s_3 \cap \bar{s_2} \cap \bar{s_1} \cap \bar{s_4}$$

$$s_4 \cap \bar{s_2} \cap \bar{s_3} \cap \bar{s_1}$$

$$s_1 \cap s_2 \cap \bar{s_3} \cap \bar{s_4}$$

$$s_3 \cap s_2 \cap \bar{s_1} \cap \bar{s_4}$$

$$s_4 \cap s_2 \cap \bar{s_3} \cap \bar{s_1}$$

$$s_1 \cap s_2 \cap s_4 \cap \bar{s_3}$$

$$s_3 \cap s_2 \cap s_4 \cap \bar{s_1}$$

$$\bar{s_1} \cap \bar{s_2} \cap \bar{s_3} \cap \bar{s_4}$$

Since the observations are independent of each other we have

$$O_{e_1,e_2,e_3,e_4,x_i} = P(S_1 = e_1, S_2 = e_2, S_3 = e_3, S_4 = e_4 | X_t = x_i)$$

$$= P(S_1 = e_1|X_t = x_i).P(S_2 = e_2|X_t = x_i).P(S_3 = e_3|X_t = x_i).P(S_4 = e_4|X_t = x_i)$$

This gives us a observation matrix **O** of size 900 x 900 for each one of the above observations.

3. **Markov assumption and Conditional independence**

   The current state is only dependent on the previous state and independent of the earlier states. i.e $P(X_t|X_{0:t-1}) = P(X_t|X_{t-1})$

   The current observation variable is dependent only on the current state and is independent of all other variables. i.e $P(E_t|E_{0:t-1}, X_{0:t}) = P(E_t|X_t)$

4. **TASKS**

   (a) **Simulation** The vehicle is simulated for 25 time steps. The actual states and the observations are recorded. The initial state of the robot is chose to the the centre of the grid. The prior probability distribution over all the states is taken to be uniform i.e $P(X_0) = [1/n.......1/n]^T$ where n is the total states (30 x 30 = 900)

   (b) **Filtering** This is the inference task of estimating the current belief state of the vehicle X_t given all the observations till that time i.e $E_{0:t}$. Basically $P(X_t \mid E_{0:t})$

   This we know is simply :

   $$P(X_t \mid E_{0:t}) = \alpha P(E_t = e_t|X_t) \sum_{x_i} P(X_t|X_{t-1} = x_i)P(X_{t-1} = x_i|e_{0:t})$$

   We can see that the above relation is a recursive formulation which consists of the transition probability, previous belief and observational probability. α is a normalising constant.

   This can be written as $P_{0:t} = \alpha * O_e \cdot T \cdot P_{0:t-1}$

   (c) **Smoothing** This is the inference task of estimating the past states given all the observations up to the present, which is $P(X_k \mid E_{0:t})$ for $0 \leq k < t$. This is computed recursively using a forward backward algorithm.

   $$P(X_{k+1}|E_{0:k}) = \sum_{x_k} P(x_{k+1}|x_k)P(x_k|e_{0:})$$

   $$P(X_k|E_{0:t}) = P(X_k|E_{0:k}) \sum_{x_{k+1}} \frac{P(X_{k+1}|X_k)P(X_{k+1}|E_{0:t})}{P(X_{k+1}|E_{0:k})}$$

   This can be written as : $S_t = \alpha P_t \cdot B_{0:t}$ and $B_{0:t-1} = T \cdot O_e \cdot B_{0:t}$

(d) **Prediction** This is the task of computing a future belief state given the observations till the present, which is $P(X_{t+k} \mid E_{0:t})$.

This is simply:

$$P(X_{t+1} \mid E_{0:t}) = \alpha P(X_{t+1}|X_t)P(X_t = x_i|e_{0:t})$$

This can be done iteratively for k steps.

This can be written as : $P_{1:t+1} = \alpha * T \cdot P_{1:t}$

(e) **Most Likely path** This is the inference task of estimating the most likely sequence of states. i.e $argmax_{x_{0:t}} P(X_{0:t}|E_{0:t})$ This is computed using the Viterbi algorithm using the following recursive relation:

$$V_{0:t} = max_{x_t}(O_e \cdot T \cdot V_{0:t-1})$$

$$x_t^{max} = argmax(V_{0:t})$$

(f) **OBSERVATIONS AND PLOTS**

For the plot for estimated log-likelihood after filtering refer to figure 1.

For the plot for estimated and ground truth locations after **filtering** refer to figure 2. The darker tracker tracks the estimated locations and the lighter one tracks the ground truth locations. The error between estimated and actual is plotted in figure 3.

For the estimated and ground truth locations after **smoothing** refer to figure 4. The darker tracker tracks the estimated locations and the lighter one tracks the ground truth locations. The error between estimated and actual is plotted in figure 5.

Observation 1: We can see that in both cases of smoothing and filtering the error (Manhattan distance) is higher in the beginning, and later on it tends to converge.

The most likely path is plotted in fig 8. The path being : [582 552 522 492 493 494 524 523 524 554 584 585 586 556 526 496 466 436 435 436 435 434 404 405 375]

$x = state//30$ $y = state\%30$

The plot for predictive likelihood after 10 time steps is plotted in figure 6 and for 25 time steps is plotted in figure 7.

Observation 2: We can see that the predictive likelihood keeps growing over the entire grid as the sensor observations are not incorporated in computing the belief. So we can say that the uncertainty in the vehicle's location increases when observations are not incorporated in calculating the belief.

We can also see from the plots that when the filtering likelihood region is small (the belief is concentrated to a small region), the error between the actual and estimated state is quite low.

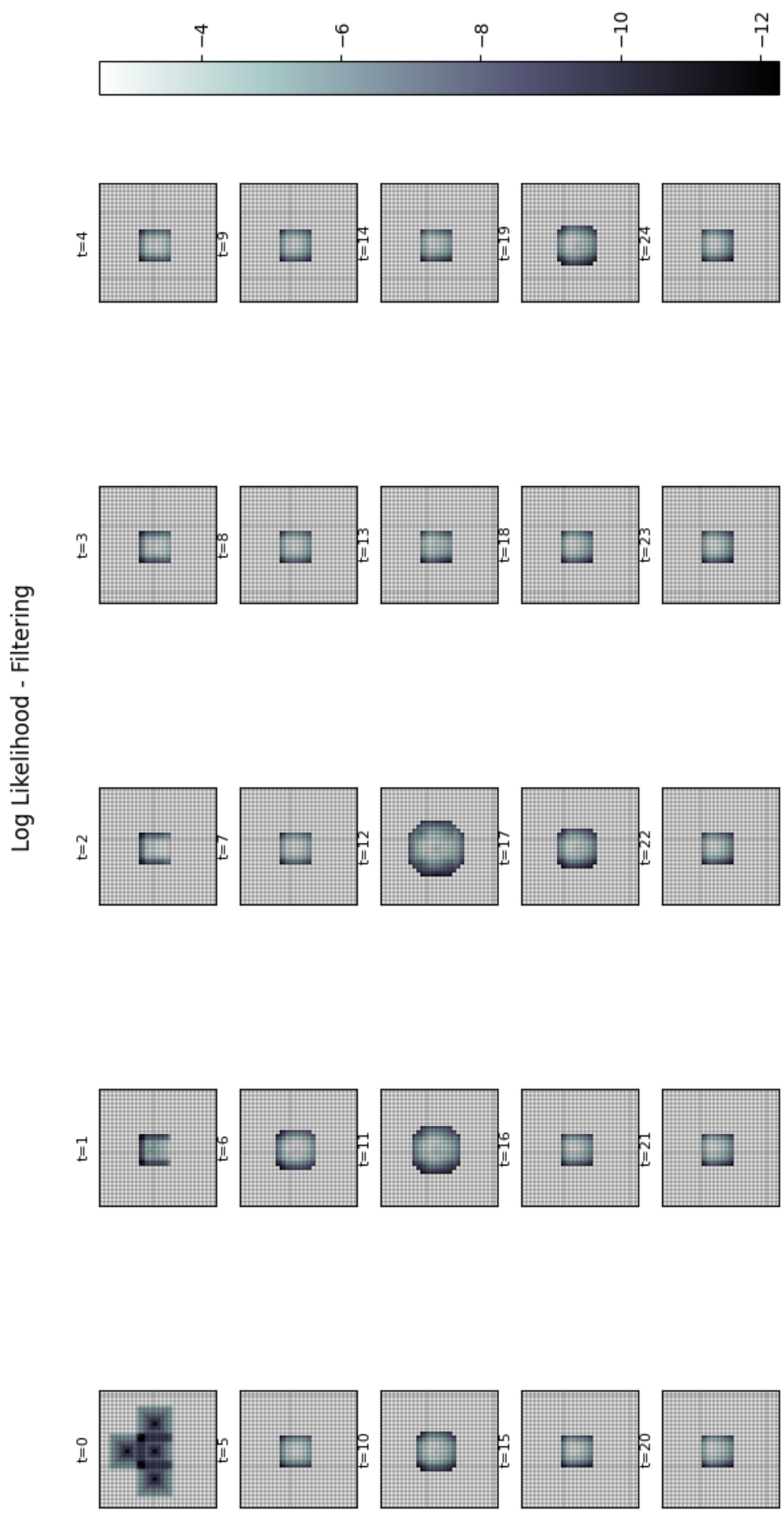
Log Likelihood - Filtering
t=0
t=1
t=2
t=3
t=4
t=5
t=6
t=7
t=8
t=9
t=10
t=11
t=12
t=13
t=14
t=15
t=16
t=17
t=18
t=19
t=20
t=21
t=22
t=23
t=24
−4
−6
−8
−10
−12

Figure 1

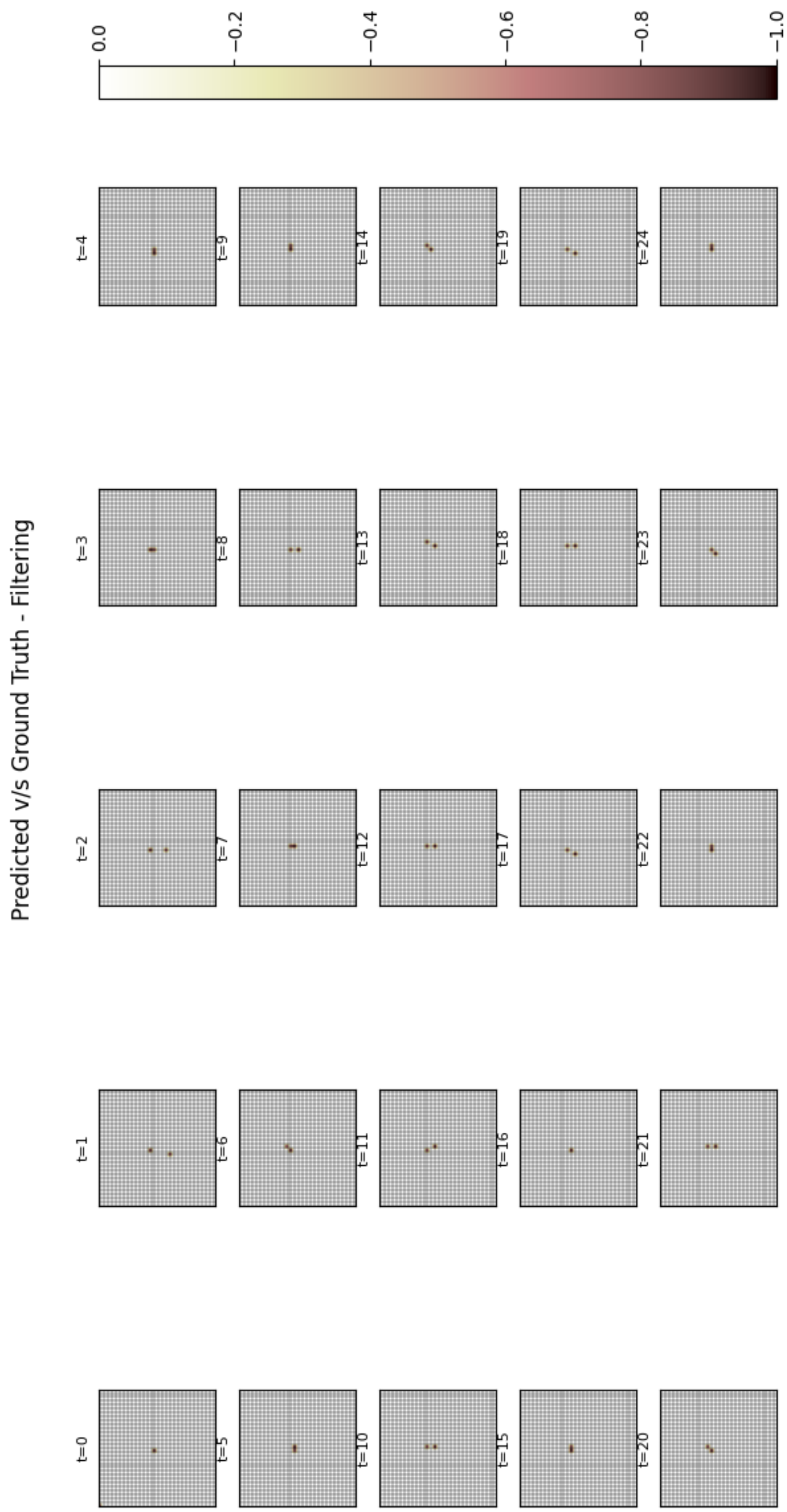
Predicted v/s Ground Truth - Filtering
t=0
t=1
t=2
t=3
t=4
t=5
t=6
t=7
t=8
t=9
t=10
t=11
t=12
t=13
t=14
t=15
t=16
t=17
t=18
t=19
t=20
t=21
t=22
t=23
t=24
0.0
−0.2
−0.4
−0.6
−0.8
−1.0

Figure 2

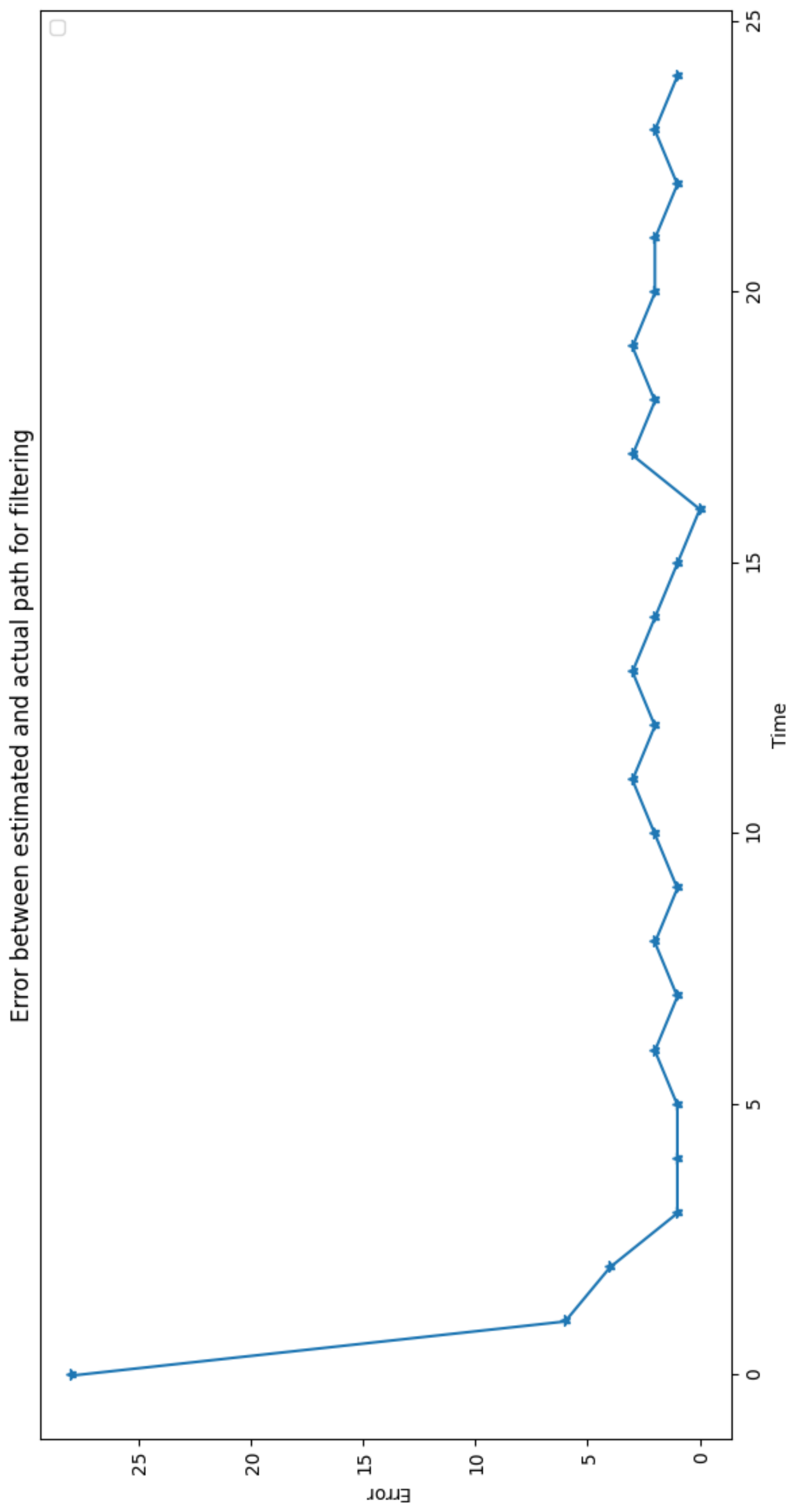
Error between estimated and actual path for filtering
Error
Time
0
5
10
15
20
25

Figure 3

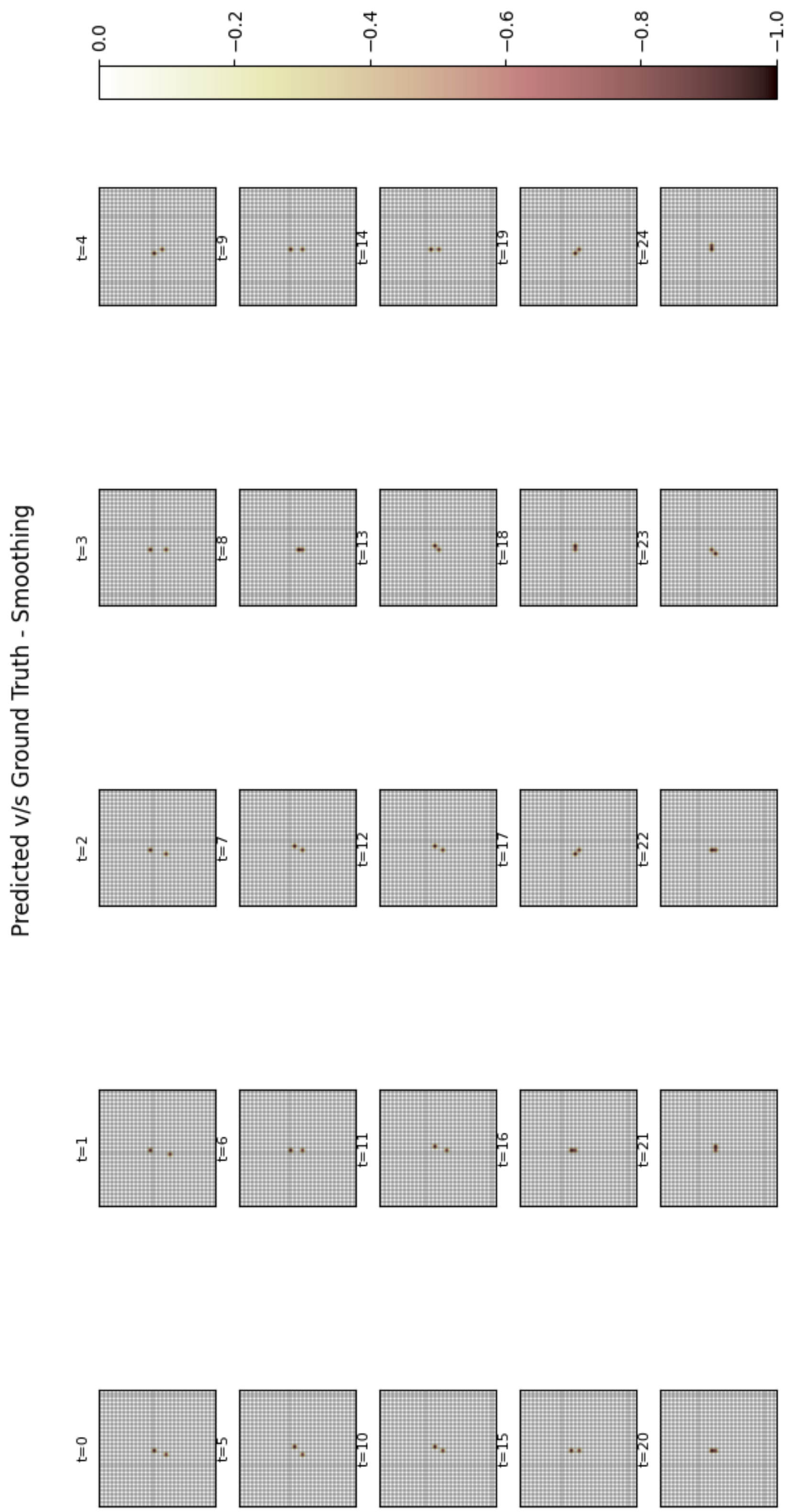
Predicted v/s Ground Truth - Smoothing
t=0
t=1
t=2
t=3
t=4
t=5
t=6
t=7
t=8
t=9
t=10
t=11
t=12
t=13
t=14
t=15
t=16
t=17
t=18
t=19
t=20
t=21
t=22
t=23
t=24
0.0
−0.2
−0.4
−0.6
−0.8
−1.0

Figure 4

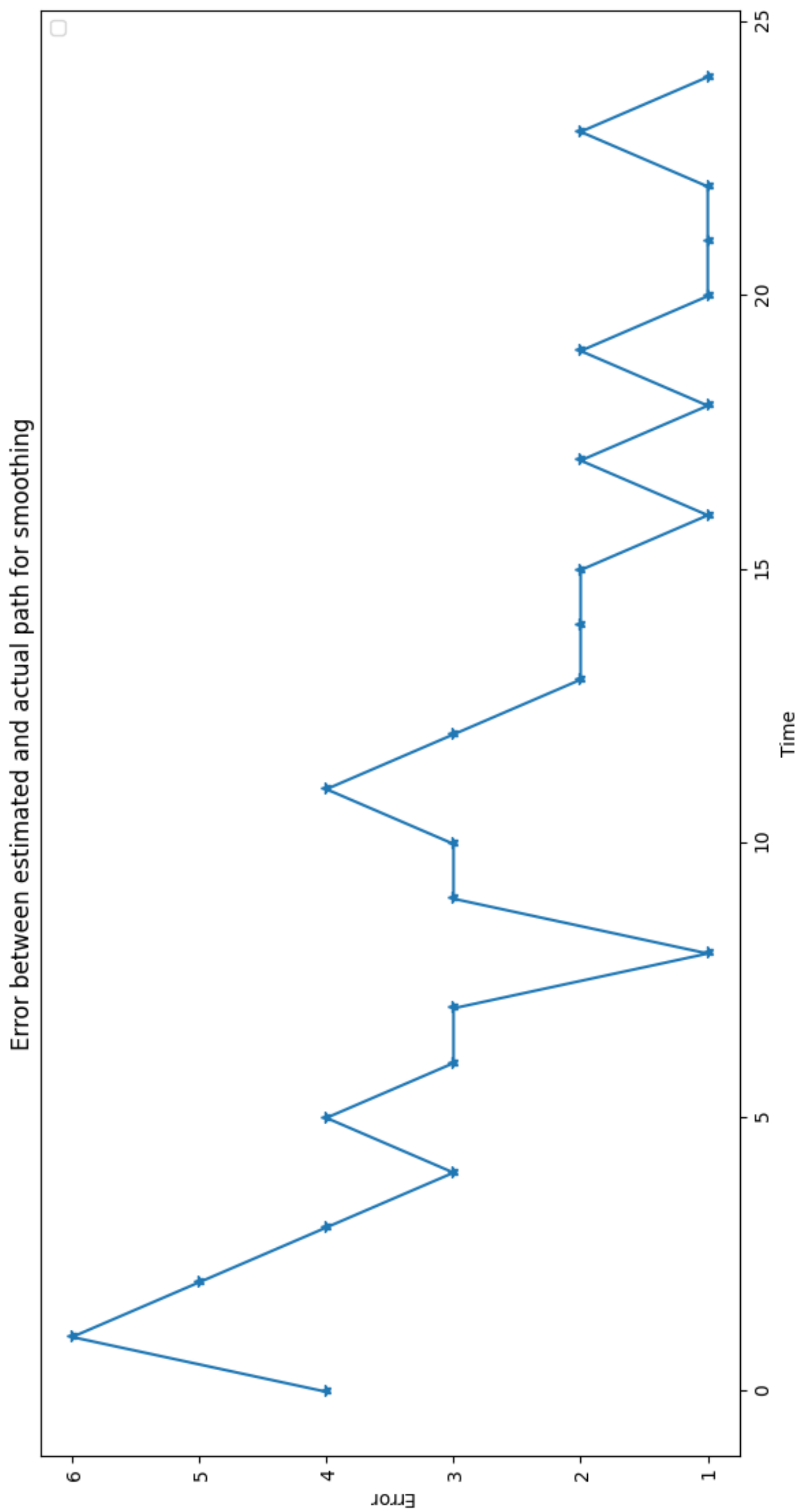
Error between estimated and actual path for smoothing
Time
Error
0
5
10
15
20
25
1
2
3
4
5
6

Figure 5

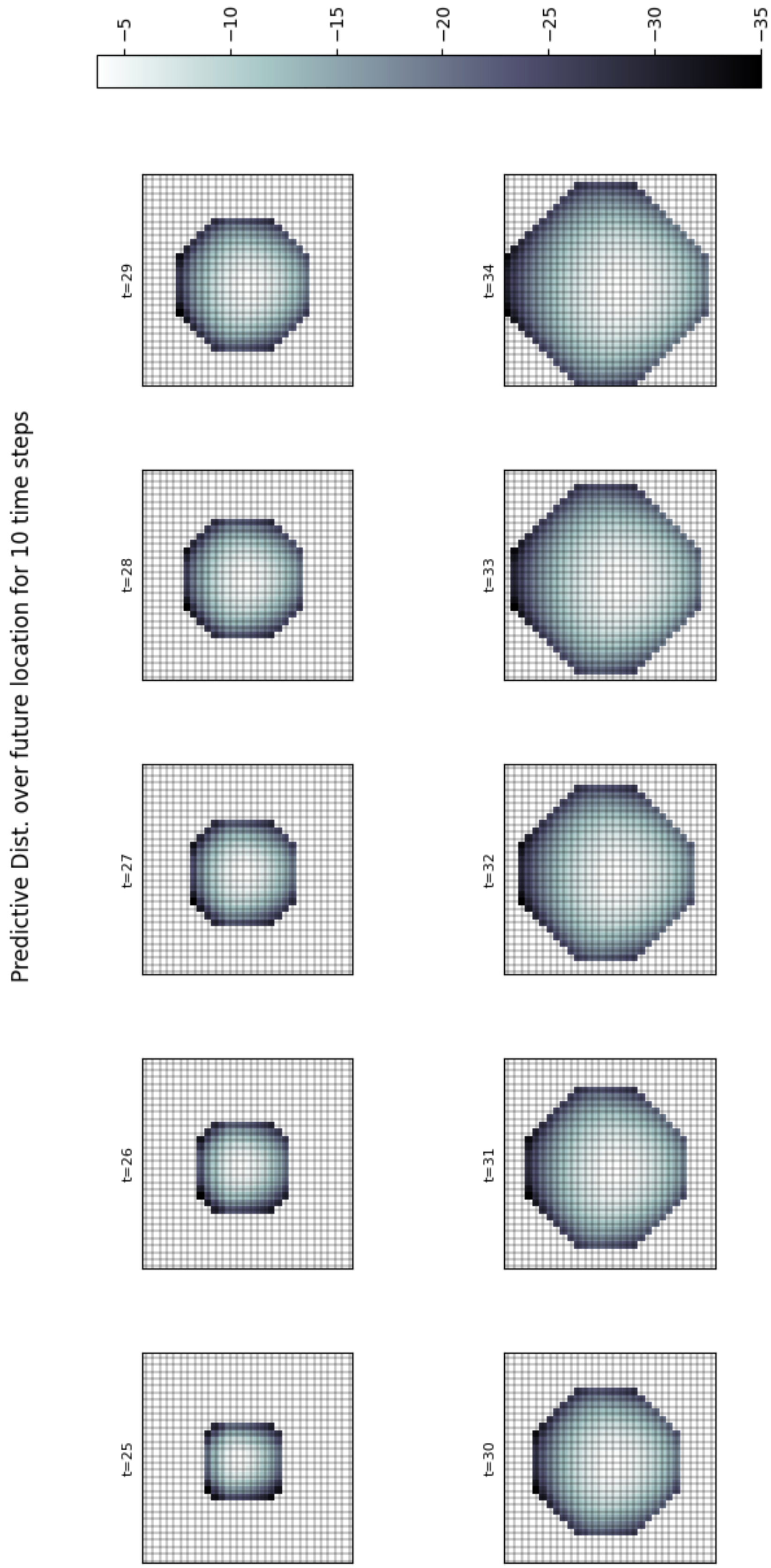
Predictive Dist. over future location for 10 time steps
t=25
t=26
t=27
t=28
t=29
t=30
t=31
t=32
t=33
t=34
−5
−10
−15
−20
−25
−30
−35

Figure 6

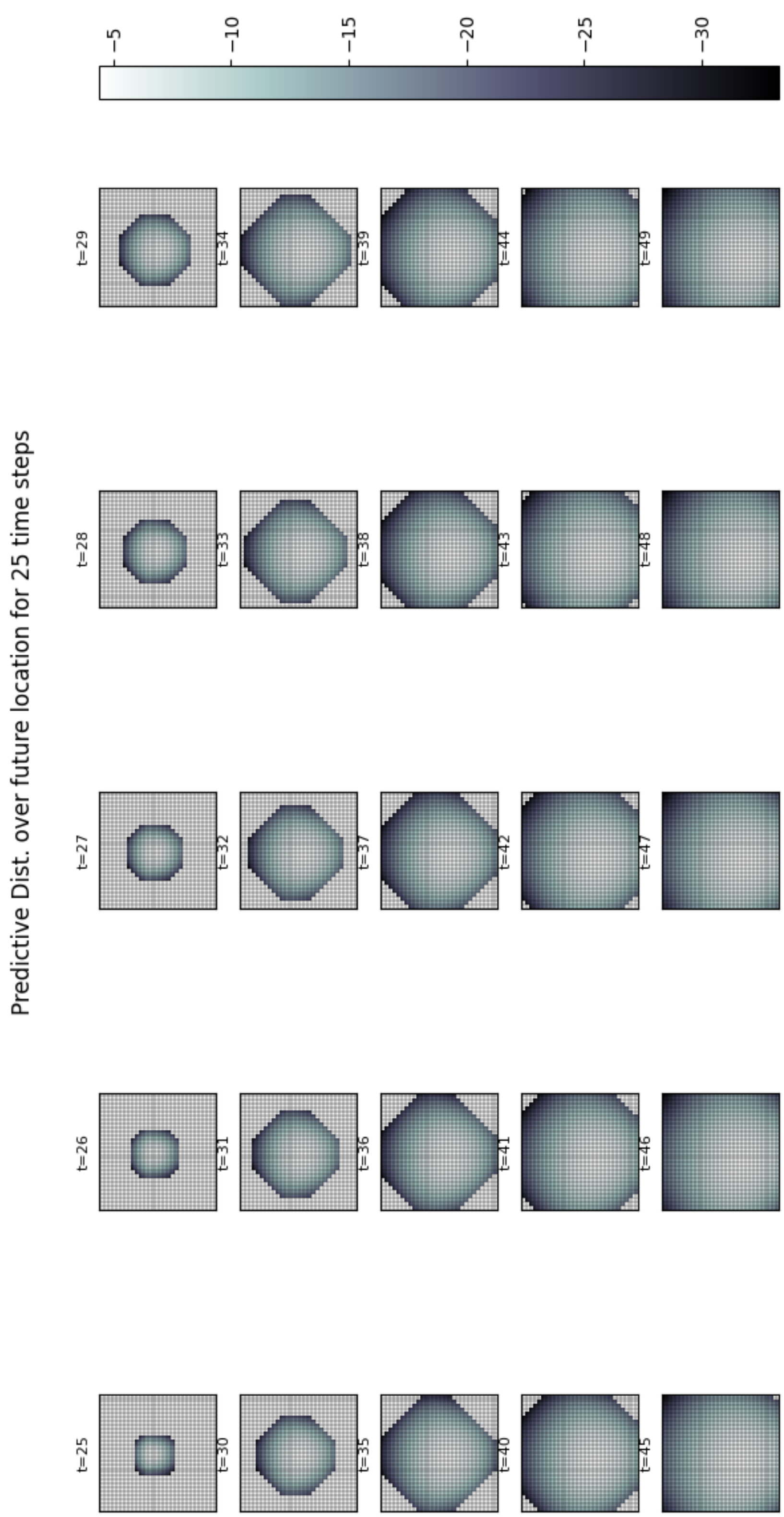
Predictive Dist. over future location for 25 time steps
t=25
t=26
t=27
t=28
t=29
t=30
t=31
t=32
t=33
t=34
t=35
t=36
t=37
t=38
t=39
t=40
t=41
t=42
t=43
t=44
t=45
t=46
t=47
t=48
t=49
−5
−10
−15
−20
−25
−30

Figure 7

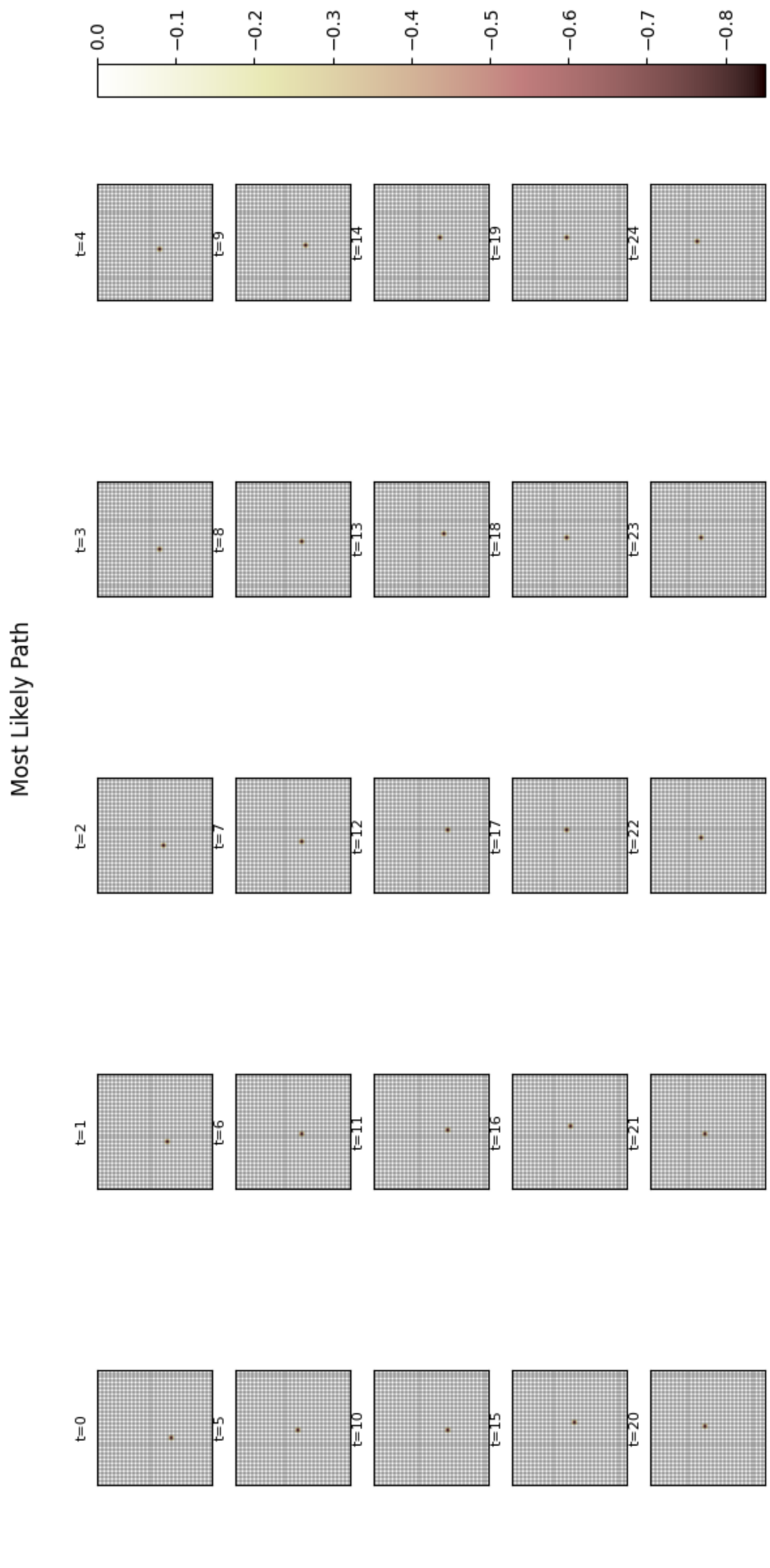
Most Likely Path
t=0
t=1
t=2
t=3
t=4
t=5
t=6
t=7
t=8
t=9
t=10
t=11
t=12
t=13
t=14
t=15
t=16
t=17
t=18
t=19
t=20
t=21
t=22
t=23
t=24
0.0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.6
−0.7
−0.8

Figure 8:

PART 2 : State Estimation for continuous variables using filters

1. **Motion model:**

$$X_t = A_t X_{t-1} + \epsilon$$

$$A_t = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \; X_t = \begin{bmatrix} x_t \\ y_t \\ v_x \\ v_y \end{bmatrix} \quad \epsilon \sim N(0, R) \quad R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0.0001 & 0 \\ 0 & 0 & 0 & 0.0001 \end{bmatrix}$$

Sensor Model:

$$Z_t = C_t X_t + \delta$$

$$C_t = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \delta \sim N(0, Q) \quad Q = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$$

The sensor and motion models are simulated for 200 time steps. If the control policy is sinusoidal then only A changes. $A_t = \begin{bmatrix} 1 & 0 & |sin(t)| & 0 \\ 0 & 1 & 0 & |cos(t)| \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

2. **Bayes Filter** The Bayesian equations for sequential state estimation updates:

For dynamic prediction:

$$P(x_t|z_{0:t-1}) = \int P(x_t|x_{t-1})P(x_{t-1}|z_{0:t-1})dx_{t-1}$$

For measurement correction:

$$P(x_t : z_{0:t}) = \frac{P(z_t|x_t)P(x_t|z_{0:t-1})}{\int P(z_t|x_t)P(x_t|z_{0:t-1})dx_{t-1}}$$

We have

$$P(x_{0:t-1}|z_{0:t-1}) = N(x_{t-1}, \mu_{t-1}, \Sigma_{t-1})$$

$$P(x_t|x_{t-1}) = N(x_t, A_{t-1}x_{t-1}, R)$$

$$P(z_t|x_t) = N(z_t, C_t x_t, Q)$$

Putting these in the update equations and solving, we get the equations that govern a **Kalman Filter**.

So we have at every step the following update:

$$\mu_{t|0:t-1} = A_t \mu_{t-1|0:t-1}$$

$$\Sigma_{t|0:t-1} = A_t \Sigma_{t-1|0:t-1} A_t^T + R$$

$$\mu_{t|0:t} = \mu_{t|0:t-1} + K_t(z_t - (C_t \mu_{t|0:t-1} + \delta))$$

$$\Sigma_{t|0:t} = (I - K_t C)\Sigma_{t|0:t-1}$$

$$K_t = \Sigma_{t|0:t-1} C^T (C \Sigma_{t|0:t-1} C^T + Q)^{-1}$$

3. **Data Association** Data association is the process of associating uncertain measurements to known tracks or latent state variables.

   Strategy: Let z be a measurement, let x be a latent state whose mean is μ and covariance matrix Σ. Define
   $$d = \sqrt{(z-\mu)^T \Sigma^{-1} (z-\mu)}$$
   . This is called the Mahalanobis distance between z and μ. We can see that d is a standard normally distributed random variable $d \sim N(0,1)$, so its square will be a chi-square distributed variable with degree of freedom 1 i.e $d^2 \sim \chi^2(1)$. Since we are simulating 2 vehicles we have 2 observations and 2 latent states. We will match the latent state with the observation using the mahalanobis distance metric. This distance metric takes into account the position of the variables, their uncertainties and also the correlations between them.

4. **OBSERVATIONS AND PLOTS**

   Observation 1: Decreasing and increasing the sensor uncertainty, doesn't have any effect on the performance of the kalman filter. This shows that the kalman filter is robust to noisy sensor measurements.

   Observation 2: We can see that the kalman filter is not able to track the velocities. This is because the kalman filter is used to only dynamically update the position measurement and it doesn't incorporate measurement from any sensor that estimates velocities.

   Observation 3: Using the data association strategy as mentioned above we were able to track the two objects efficiently. I also tested for different values of sensor and motion uncertainties and also for different initial positions, and it works really well. There are very less instances of misjudgement of assigning the wrong observation to a particular latent state. Since it works well for 2 objects I feel that we can scale this up 4-5 objects. This is a non Bayesian technique of association, there could be certain Bayesian techniques that perform better than this.

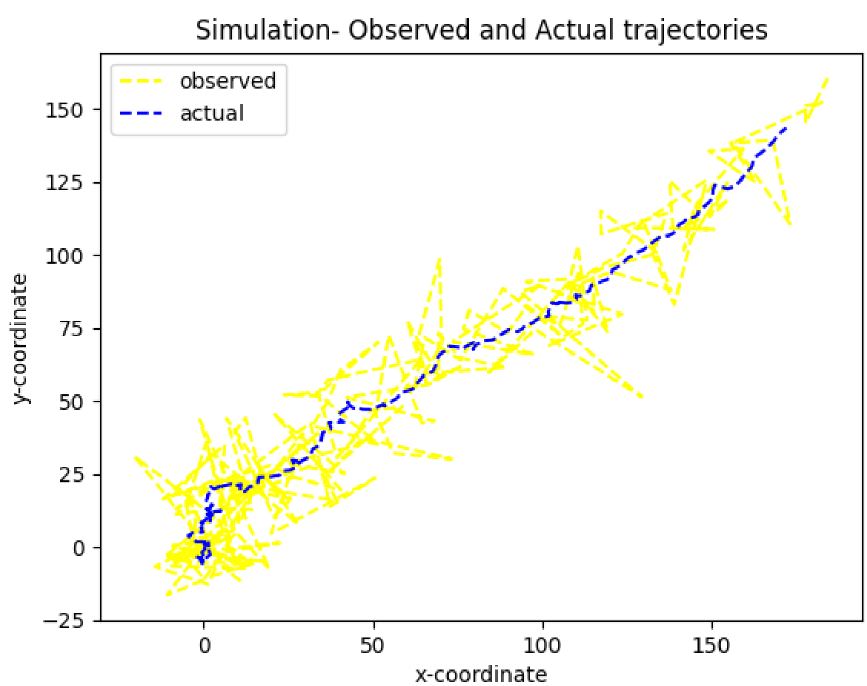

Figure 1 : Simulation for 200 time steps

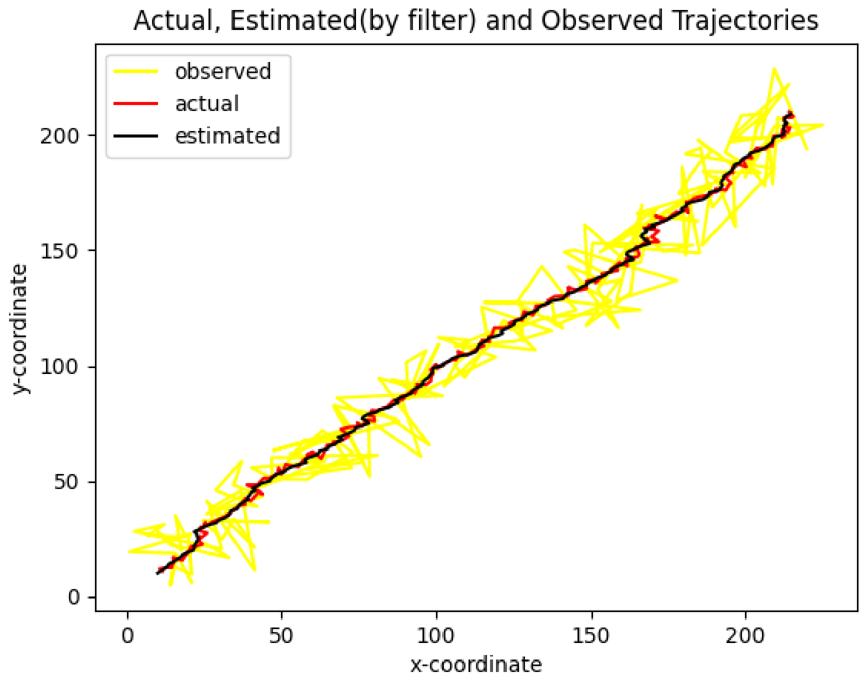

Figure 2 : Estimation by filtering

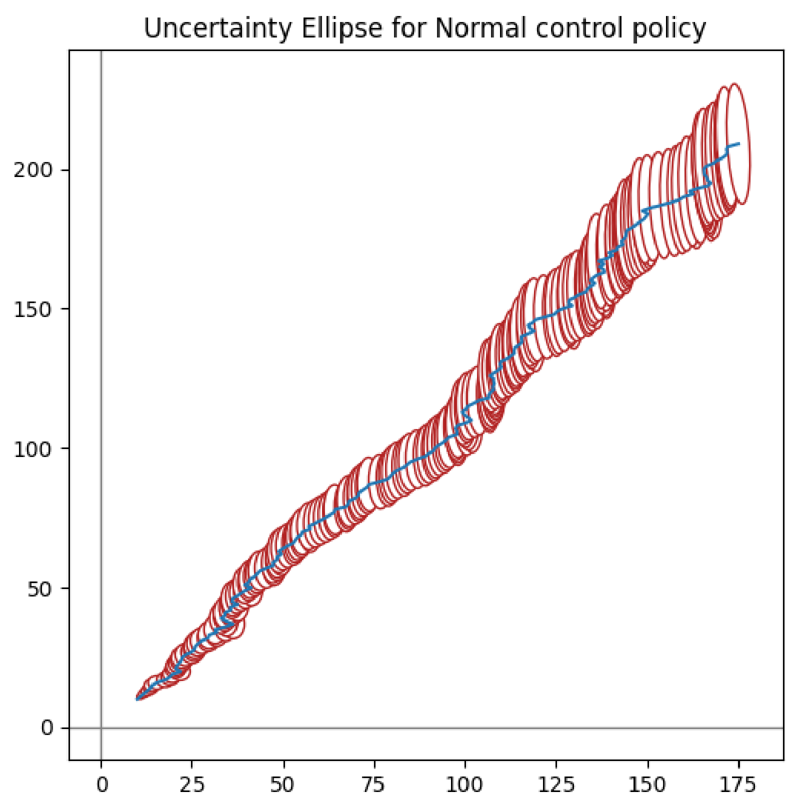

Figure 3 : The uncertainty ellipses for the above estimation. We can see that the radii of the ellipse increase which tells us about the evolution of uncertainty as we progress in time.

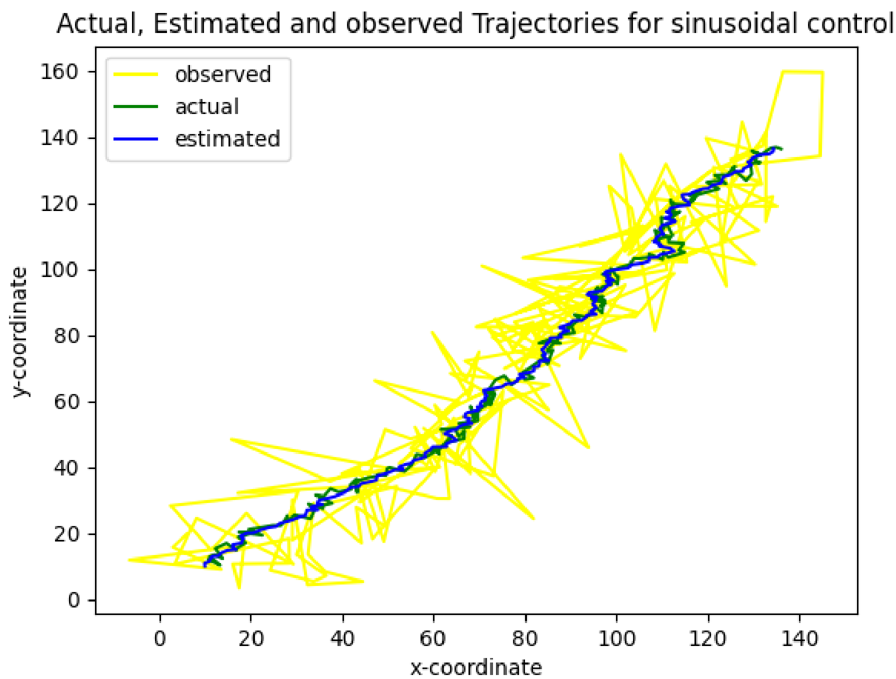

Figure 4 : Estimation using sinusoidal control policy.

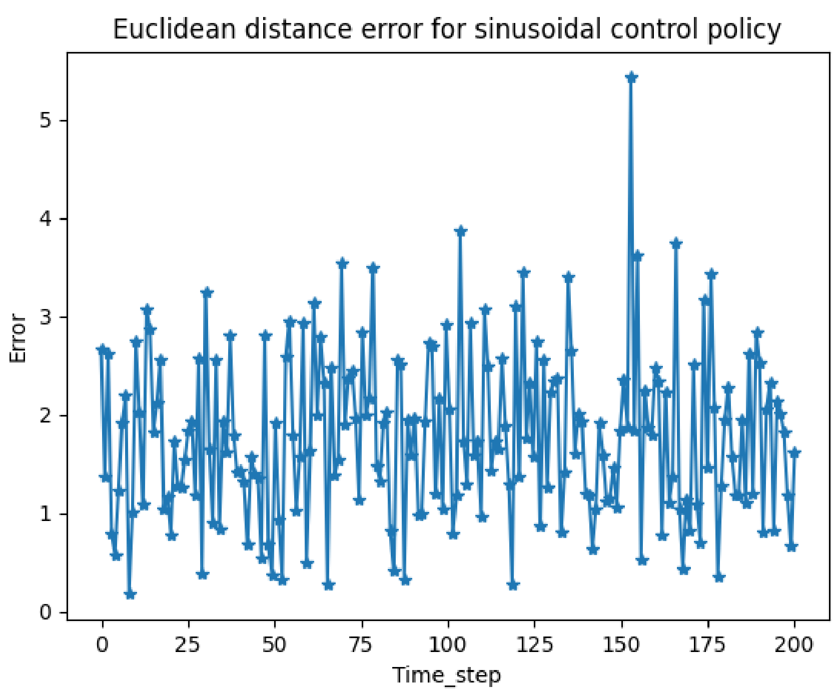

Figure 5 : Euclidean Error between actual and estimated trajectory for the above estimation. There is no particular pattern involved.

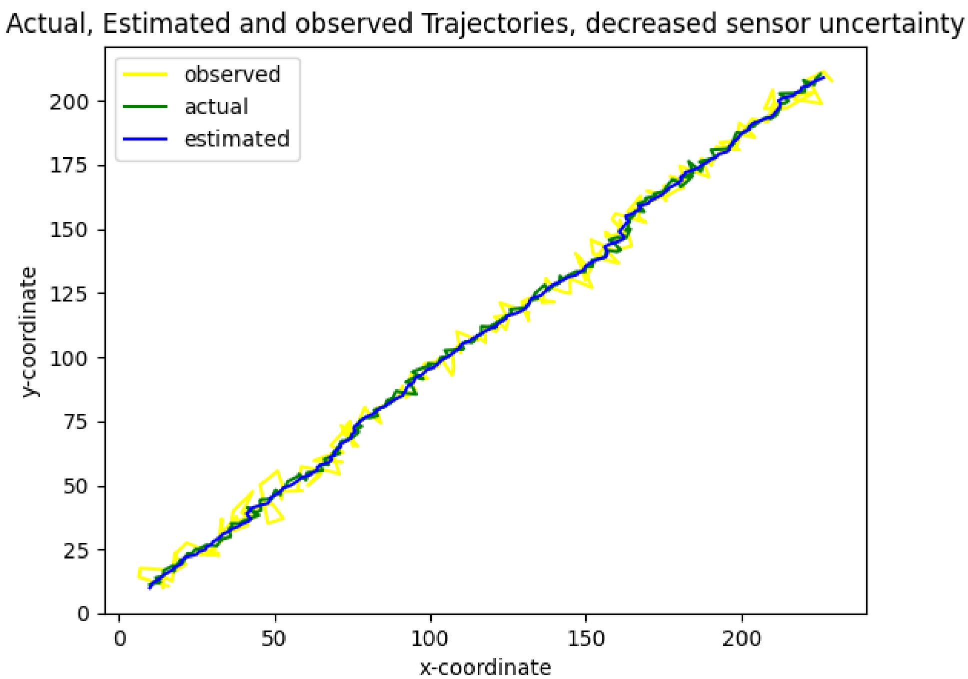

Figure 6 : Since the uncertainty in sensor is decreased we can see that the observation noise has decreased, without much effect on the performance of the filter.

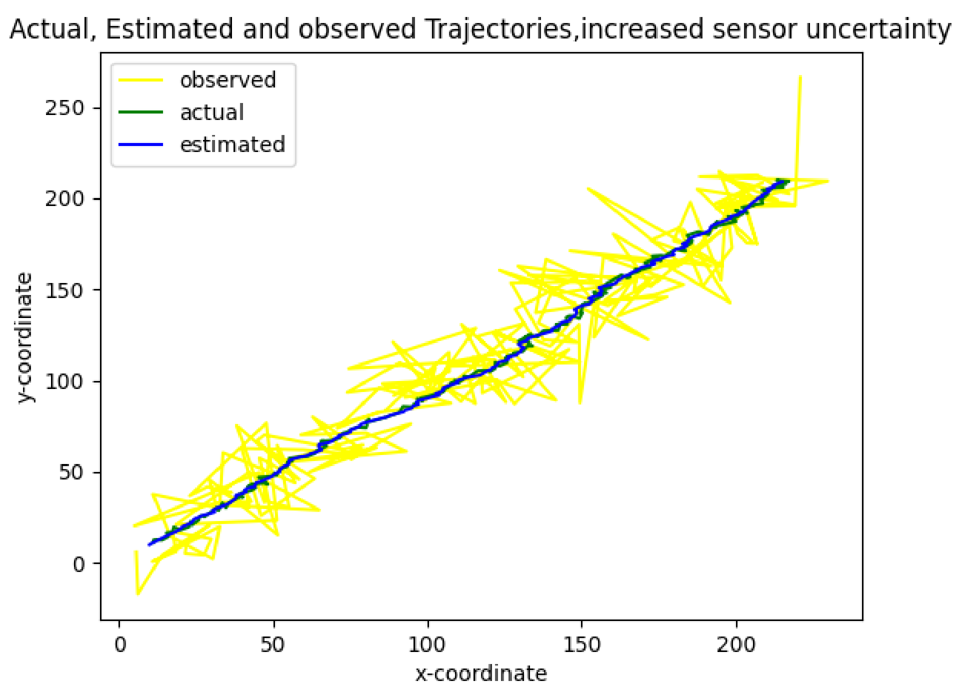

Figure 7 : Since the uncertainty in sensor is increased we can see that the observations have become more noisy, without much effect on the performance of the filter.

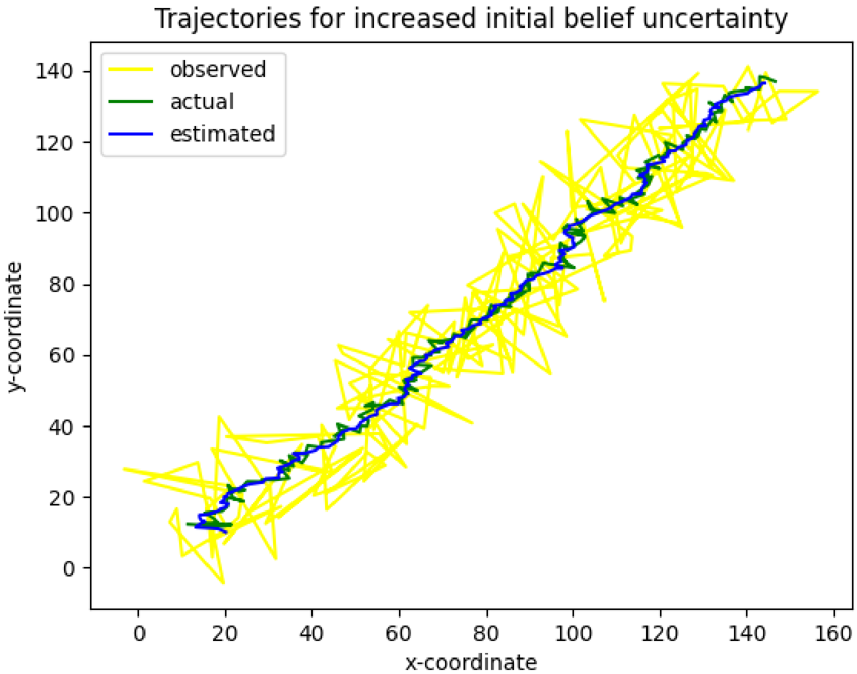

Figure 8 : Even with increase in initial belief uncertainty we can see that that there is not much effect on the performance of the filter.

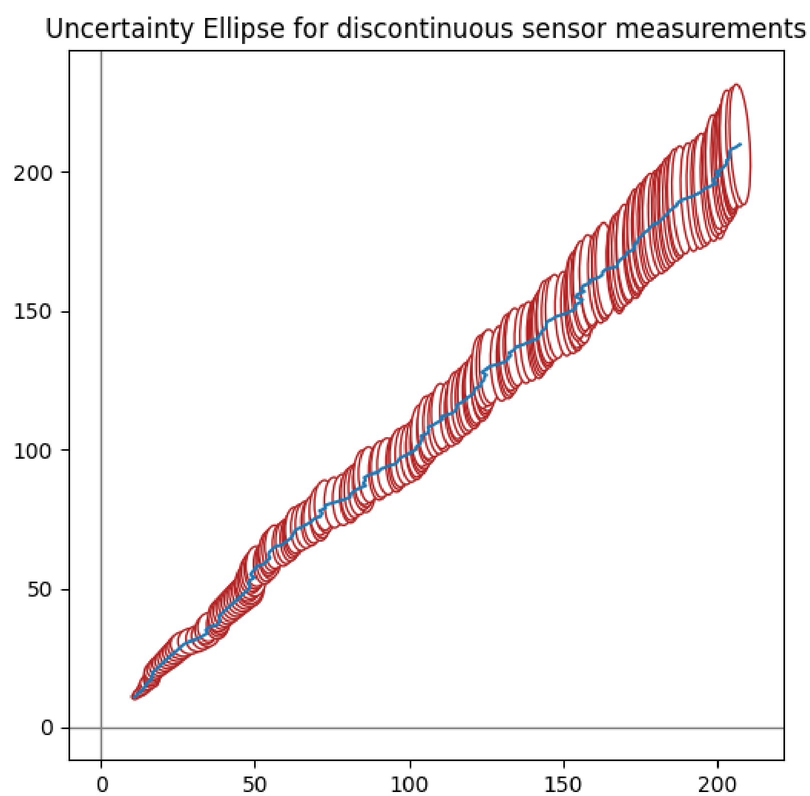

Figure 9 : Even when the sensor measurements are discontinuous from t=10-20 and also from t=30-40, there isn't much variation in the estimated path.

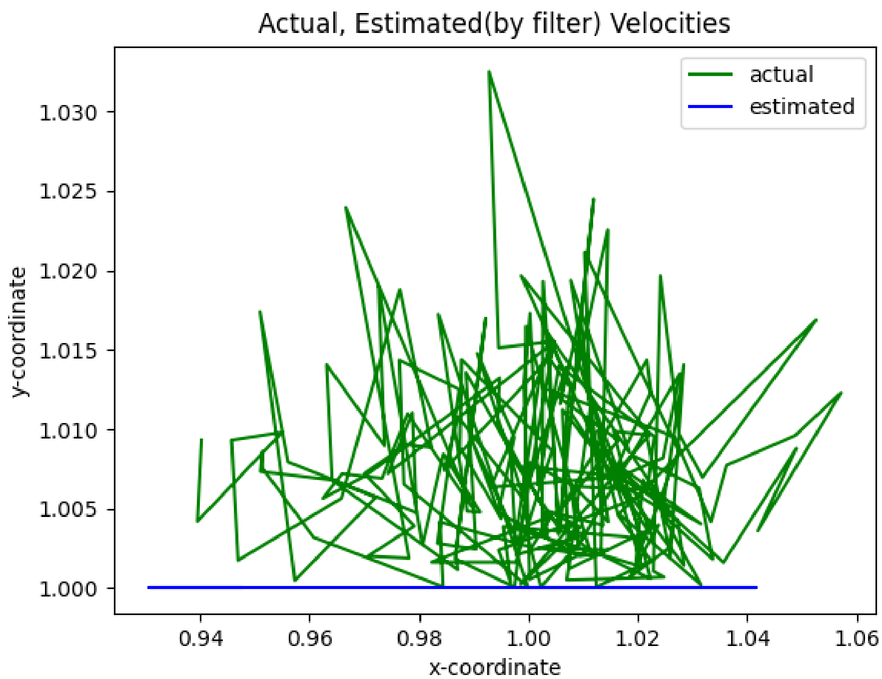

Figure 10 : We can see that the filter cannot track the velocities.

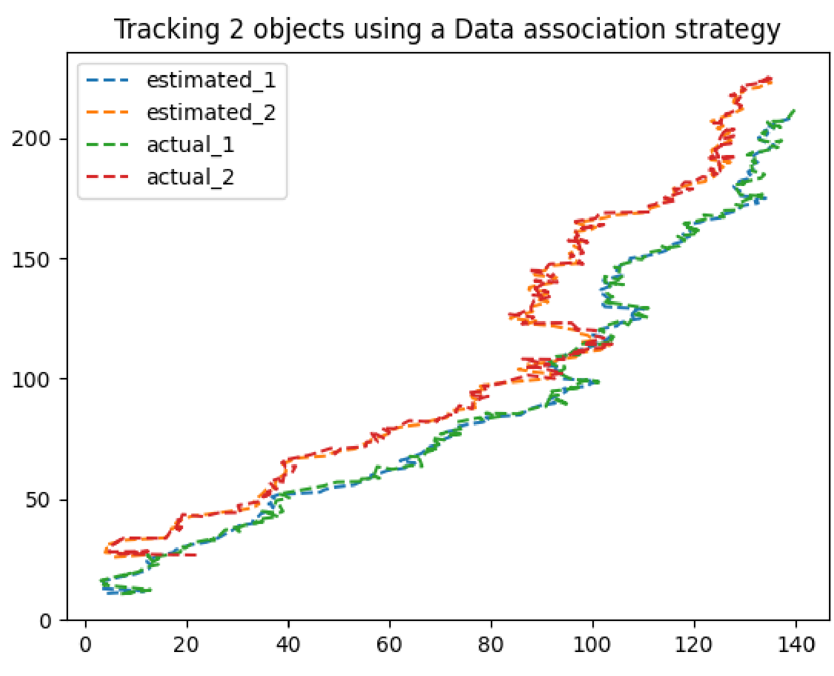

Figure 11 : Data association for tracking 2 objects. The actual and estimated paths for both the objects almost overlap, and this has been tried out for many initial positions of the objects.

References

[1] Artificial Intelligence: A modern approach; Stuart J. Russell and Peter Norvig

[2] Probabilistic Robotics; Thrun, Sebastian, Wolfram Burgard, and Dieter Fox.

[3] Class slides on state estimation

[4] http://ais.informatik.uni-freiburg.de/teaching/ws09/robotics2/pdfs/rob2-11-dataassociation.pdf