16-833: Robot Localization and Mapping, Spring 2023

Homework 1
SLAM using Extended Kalman Filter (EKF-SLAM)

Prakrit Tyagi

March 18 2023

1 Theory

The pose of the robot in the global coordinates at time t is written as a vector

$$\mathbf{p}_t = \begin{bmatrix} x_t & y_t & \theta_t \end{bmatrix}^\top \tag{1}$$

where x_t and y_t are the 2D coordinates of the robot's position, and θ_t is the robot's orientation

1.1 Predicting the pose

Predicting the next pose P_{t+1} as a nonlinear function of the current pose pt and the control inputs d_t and α_t

$$x_{t+1} = x_t + d_t \cos\theta_t \tag{2}$$

$$y_{t+1} = y_t + d_t \sin\theta_t \tag{3}$$

$$\theta_{t+1} = \theta t + \alpha_t \tag{4}$$

$$P_{t+1} = \begin{pmatrix} x_{t+1} \\ y_{t+1} \\ \theta_{t+1} \end{pmatrix} \tag{5}$$

1.2 Predicting the Uncertainty

We are given the errors that follow Gaussian distributions in x-direction, y-direction and in rotation respectively.

$$e_x \sim \mathcal{N}\left(0, \sigma_x^2\right) \tag{6}$$

$$e_y \sim \mathcal{N}\left(0, \sigma_y^2\right) \tag{7}$$

$$e_\alpha \sim \mathcal{N}\left(0, \sigma_\alpha^2\right) \tag{8}$$

This follows that the process noise associated with each pose element as:

$$R_t = \begin{pmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_\alpha^2 \end{pmatrix} \tag{9}$$

The co-variance associated with the pose can be given by

$$\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t \tag{10}$$

where A_t is given by linearizing the prediction equation given in 1.1 with respect to P_t

$$A_t = \frac{\partial P_{t+1}}{\partial P_t} = \begin{pmatrix} 1 & 0 & -d_t \sin\theta \\ 0 & 1 & d_t \cos\theta \\ 0 & 0 & 1 \end{pmatrix} \tag{11}$$

1.3 Landmark estimation

We are given the measurement in terms of the bearing angle β (in the interval $(\pi, \pi]$) and the range r with noise $n_\beta \sim \mathcal{N}\left(0, \sigma_\beta^2\right)$ and $n_r \sim \mathcal{N}\left(0, \sigma_r^2\right)$ then the estimation of landmark postions is:

$$l_x = x_t + (r + n_r)\cos\left(n_\beta + \beta + \theta_t\right) \tag{12}$$

$$l_y = y_t + (r + n_r)\sin\left(n_\beta + \beta + \theta_t\right) \tag{13}$$

1.4 Measurement estimation

We are given the position of the landmark (l_x, l_y) in global coordinates, P_t and the noise terms

$$\begin{pmatrix} \beta \\ r \end{pmatrix} = \begin{pmatrix} wrap2pi(atan2\left(\left(l_y - y\right), \left(l_x - x\right)\right) - \theta_t - n_\beta) \\ \sqrt{\left(l_x - x\right)^2 + \left(l_y - y\right)^2} - n_r \end{pmatrix} \tag{14}$$

1.5 Jacobian H_p

To find the measurement Jacobian we need to take the jacobian of equation 14 with respect to P_t

$$H_p = \begin{pmatrix} \delta_y/d & -\delta_x/d & -1 \\ -\delta_x/\sqrt{d} & -\delta_y/\sqrt{d} & 0 \end{pmatrix} \tag{15}$$

where

$$d = \left(l_x - x\right)^2 + \left(l_y - y\right)^2 \tag{16}$$

$$\delta_x = l_x - x \tag{17}$$

$$\delta_y = l_y - y \tag{18}$$

1.6 Jacobian H_l

To find the measurement Jacobian we need to take the jacobian of equation 14 with respect to l_x and l_y

$$H_l = \begin{pmatrix} -\delta_y/d & \delta_x/d \\ \delta_x/\sqrt{d} & \delta_y/\sqrt{d} \end{pmatrix} \tag{19}$$

Why do we not need to calculate the measurement Jacobian with respect to other landmarks except for itself (Hint: based on what assumption)?

Answer: We do this because we assume that the landmark poses are independent of each other.

2 Implementation and Evaluation

2.1 Total Landmarks

The total landmarks are 6. The data file give 12 measurement values at each time step which corresponds to bearing and range for each landmark.

2.2 Final Trajectory and co variances figure 1

The final trajectory obtained can be seen in the figure 1.

2.3 Describe how EKF-SLAM

The Extend Kalman Filtering based SLAM works on two basic steps which are prediction and correction.

When the robot moves and a prediction is made about its pose, it increases the the uncertainty of the robot's position. This is because the motion performed will not be exactly as directed by the controls. This can be seen in the figure 2 where the Magenta color circles are uncertainties induced by the robot motion. Moving forward when the measurement is received and correction is made to the previous prediction, the uncertainty is decreased(blue circle). This is done by comparing the uncertainty in the measurement and the motion. Mathematically the Kalman Gain tells us by

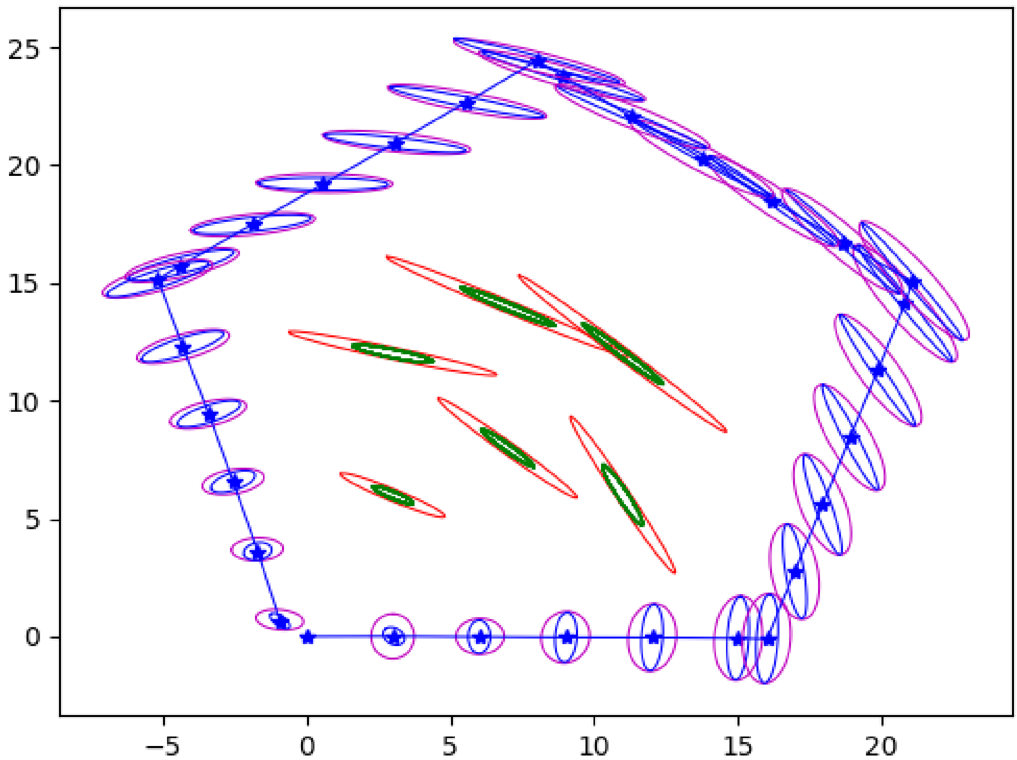

Figure 1: The Final trajectory and the landmarks with their co variances in the shape of 3-σ ellipses

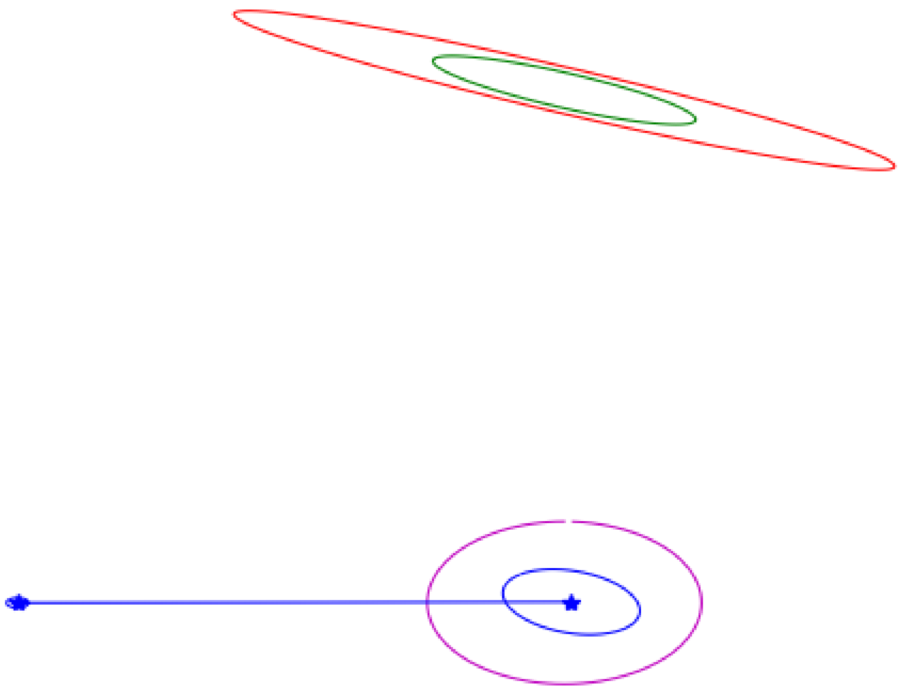

Figure 2: EKF single step

what factor should we trust the measurement over the motion. Generally the measurement has less uncertainty and therefore helps in reducing the uncertainty as seen by the blue color circle around robot.

This same principle applies to the pose estimation of the landmarks by the Kalman filter.

2.4 Evaluation of Output Map

The Landmark positions can be seen as small solid circles inside the figure 3. We can see that the Landmark postions are inside their corresponding co variance circles and are at the center of the co variance circles. This tells us that our EKF slam was able to positively estimate the position of landmarks as close to their true values with small uncertainty.

The Euclidean and Mahalanobis distances of each landmark estimation with respect to the ground truth is given in the table 1

The Mahalanobis distance can tell us how much far away is the true center of the landmarks from the distribution in terms of the standard deviation This is a performance measure which will tell us the quality of our distribution. We can tell this in terms of standard deviation by first calculating the standard deviation by dividing the Euclidean by the Mahalanobis distance. For example take values for landmark 1.

$$\sigma_1 = \frac{Euclidean}{Mahalanobis} = \frac{0.00536749}{0.05859441} = 0.09160413083 \tag{20}$$

So now we can say that the true center of the landmark 1 is 0.05859441 standard deviations σ_1 away from the distribution center(mean).

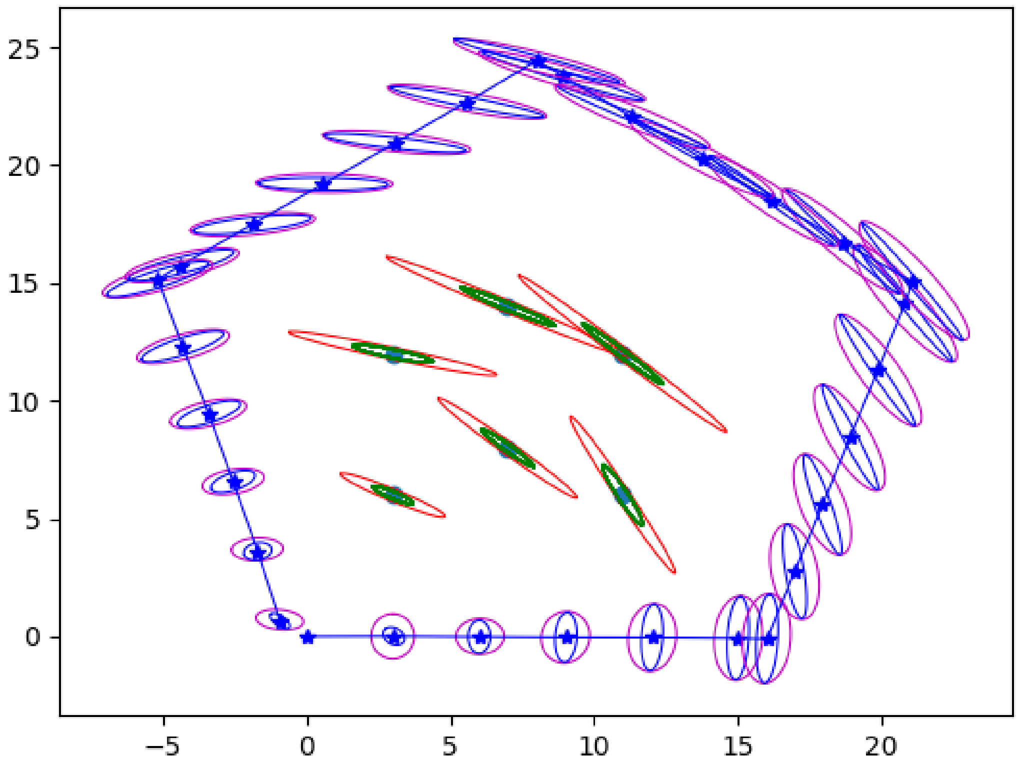

Figure 3: Landmarks True position plotted on the Output figure

Euclidean	Mahalanobis
0.00536749	0.05859441
0.01366047	0.07145138
0.00711428	0.04048998
0.01295886	0.06883512
0.00992717	0.027285995
0.01586617	0.097660937

Table 1: Parameter table

3 Discussion

3.1 Final Covariance matrix

- The initial landmark co variance matrix has zeros in off diagonal because we made an assumption that the landmarks are independent of each other and knowing one landmarks position has no affect on any other landmarks position. This changes as the simulation move on, as the Kalman gain which updates the full covariance matrix becomes a full matrix.
- When setting the full covariance matrix certain assumptions are made. We assume that the robot pose and the landmarks are not correlated. Due to which there are zeros in co variance matrix at positions that are cross related to robot and landmarks.

3.2 Other Results

3.2.1 Increasing the σ_x and σ_y10x

1. When the standard deviation in x position of the robot is increased by a factor of 10 the co variance circle greatly increases in the x direction as seen in the figure 4.
2. We can see that the co variance in the y direction also increases in some time steps suggesting that there is some correlation between the x y poses of the robot.
3. Though at the end of simulation when the robot is localized the y axis covariance is greatly reduced. Which tells us that the assumptions we made regarding independence between x y and theta is justified for initialization of P of robot.

Similarly we can see the same behaviour when the standard deviation of robot's y position is increased a factor of 10 in the figure 5.

3.2.2 Increasing the σ_α 10x

Increasing the standard deviation of α by a factor of 10 does'nt show much difference when compared with original output. The figure 6 shows the results

3.2.3 Decreasing the σ_β 10

Decreasing the standard deviation of bearing angle β by a factor of 10 does'nt show much difference when compared with original output. The figure 7 shows the results

3.2.4 Increasing the σ_r 10x

Increasing the standard deviation of the range measurement by a factor of 10 greatly affects the co variances of landmarks. We can observe in the figure 8 the co variances of the landmarks have become big after all the updates when compared to the original output.

3.3 Imporovement suggestions for performance enhancement

- Keeping a value function that calculates the value of a landmark on the bases of certain parameters such as distance from the robot and previous co variance. This can help us in selection of best N landmarks from a greater set. This will allow us to only update N landmarks at each iteration

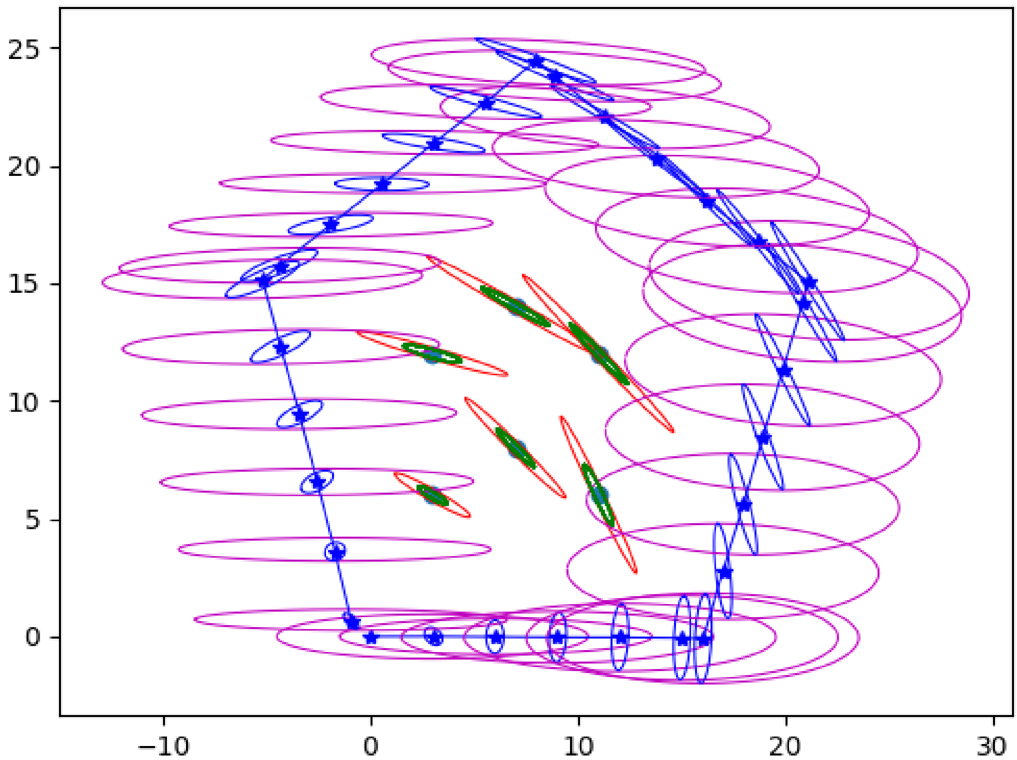

Figure 4: Increasing the standard deviation of robot x postion by a factor of 10

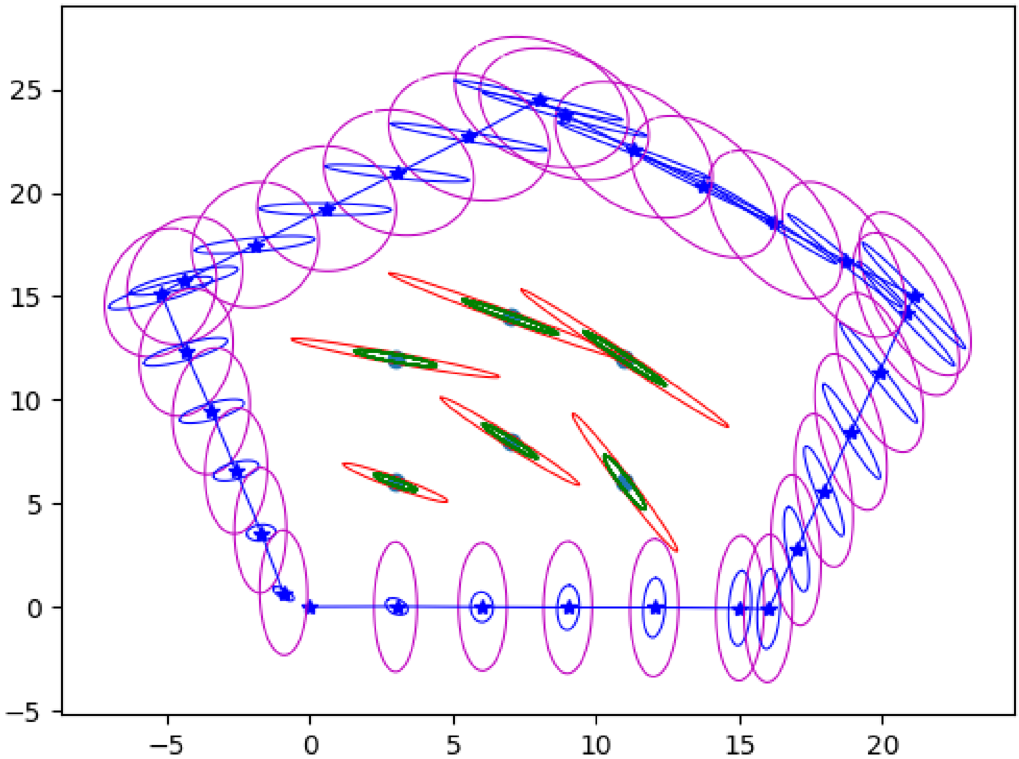

Figure 5: Increasing the standard deviation of robot y postion by a factor of 10

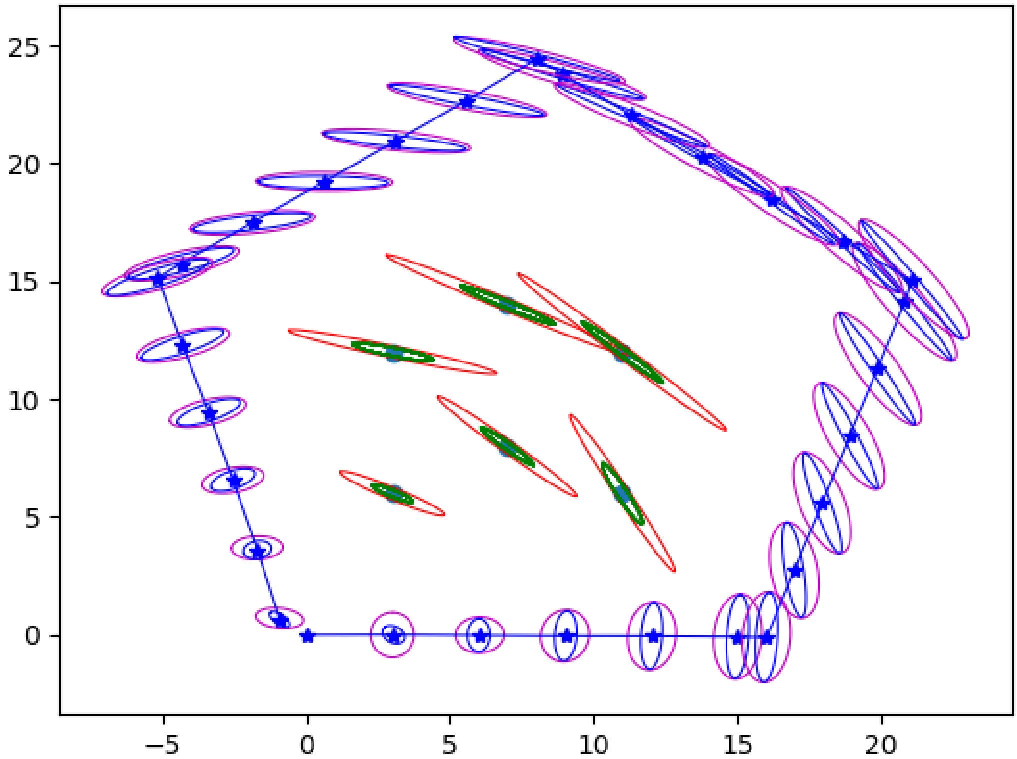

Figure 6: Increasing the standard deviation of robot α postion by a factor of 10

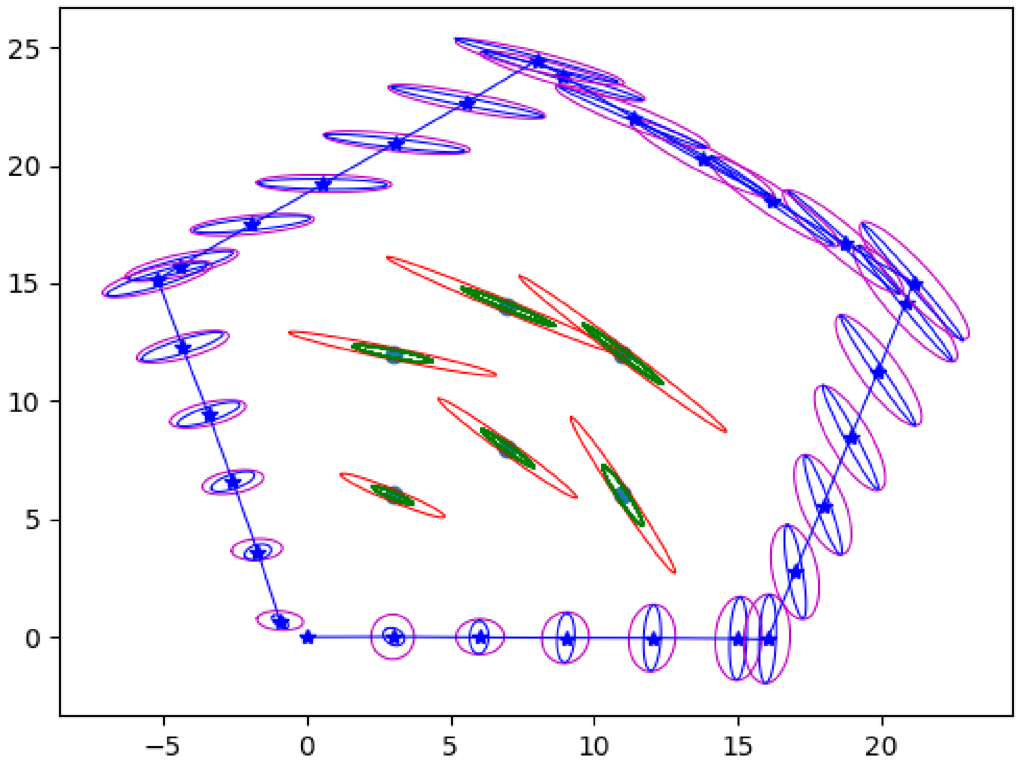

Figure 7: Decreasing the standard deviation of bearing angle β by a factor of 10

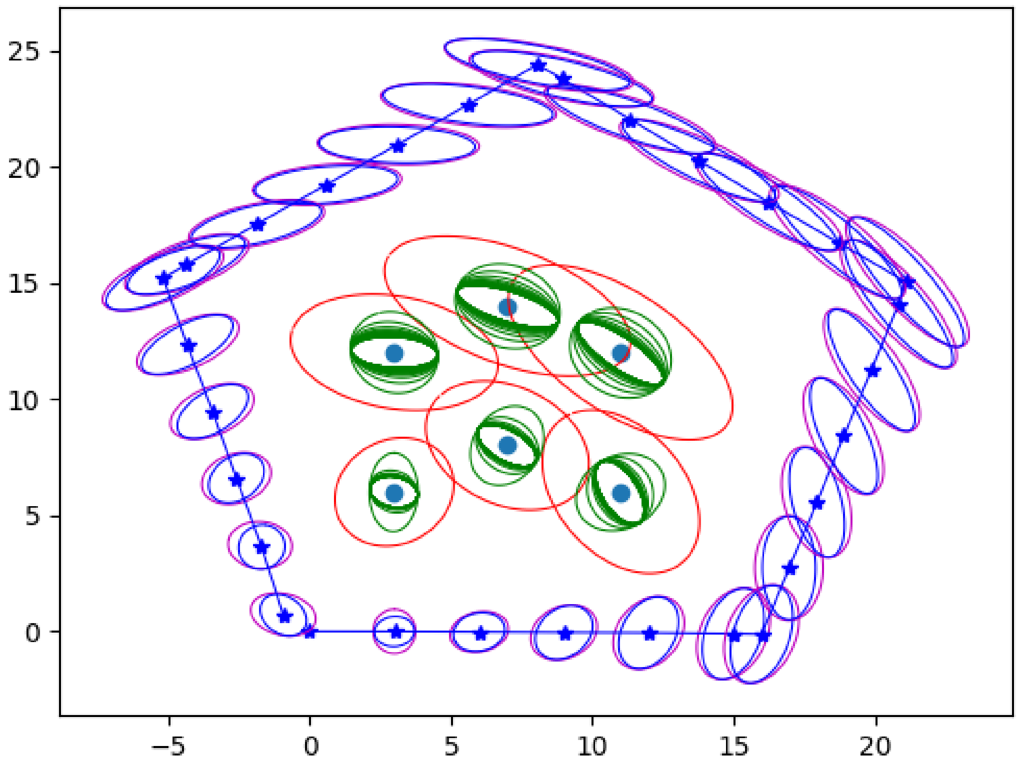

Figure 8: Decreasing the standard deviation of range measuremnt by a factor of 10

- Random sampling of fixed set of landmarks keeps us away from value function.