Part 1

Qishun Yu

1. $$P_t = \begin{bmatrix} x_t \\ y_t \\ \theta_t \end{bmatrix}$$

$$P_{t+1} = \begin{bmatrix} x_t + d_t \cdot \cos(\theta_t) \\ y_t + d_t \cdot \sin(\theta_t) \\ \theta_t + \alpha_t \end{bmatrix}$$

2.

$$\boxed{\Sigma_{t+1} = G_{t+1}\,\Sigma_t\, G_{t+1}^{T} + R \cdot U_{t+1} \cdot R^{T}}$$

$$G_{t+1} = \frac{\partial g}{\partial x_{t+1}}$$

$$= \begin{bmatrix} 1 & 0 & -d_{t+1}\sin(\theta_t) \\ 0 & 1 & d_{t+1}\cos\theta_t \\ 0 & 0 & 1 \end{bmatrix}$$

Uncertainty in Robot frame

$$U_{t+1} \sim N\left(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_\alpha^2 \end{bmatrix} \right)$$

Uncertainty in Global frame

$$\sim \begin{bmatrix} \cos\theta_t & -\sin\theta_t & 0 \\ \sin\theta_t & \cos\theta_t & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_\alpha^2 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & \sin\theta & \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

3.

$$\begin{bmatrix} l_x \\ l_y \end{bmatrix} = \begin{bmatrix} P_{tx} + (r + n_r) \cdot \cos(\beta + n_\beta + P_{t\theta}) \\ P_{ty} + (r + n_r) \cdot \sin(\beta + n_\beta + P_{t\theta}) \end{bmatrix}$$

4.

$$r = \sqrt{(lx - P_{tx})^2 + (ly - P_{ty})^2} + n_r$$

$$\beta = wrap2pi(np.arctan2(ly - P_{ty}, lx - P_{tx}) + n_\beta - P_{t\alpha})$$

$$P = \begin{bmatrix} \beta \\ r \end{bmatrix} = \begin{bmatrix} wrap2pi(np.arctan2(ly - P_{ty},\ lx - P_{tx}) - P_{t\alpha}) \\ \sqrt{(lx - P_{tx})^2 + (ly - P_{ty})^2} \end{bmatrix}$$

$$C = \begin{bmatrix} \sigma_\beta^2 & 0 \\ 0 & \sigma_r^2 \end{bmatrix}$$

5.

$$H_p = \begin{bmatrix} \frac{\partial h_\beta}{\partial P_{tx}} & \frac{\partial h_\beta}{\partial P_{ty}} & \frac{\partial h_\beta}{\partial P_{t\alpha}} \\ \frac{\partial h_r}{\partial P_{tx}} & \frac{\partial h_r}{\partial P_{ty}} & \frac{\partial h_r}{\partial P_{t\alpha}} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{ly - P_{ty}}{(lx - P_{tx})^2 + (ly - P_{ty})^2} & \frac{lx - P_{tx}}{(lx - P_{tx})^2 + (ly - P_{ty})^2} & -1 \\ \frac{P_{tx} - lx}{\sqrt{(lx - P_{tx})^2 + (ly - P_{ty})^2}} & \frac{P_{ty} - ly}{\sqrt{(lx - P_{tx})^2 + (ly - P_{ty})^2}} & 0 \end{bmatrix}$$

6.

$$H_t = \begin{bmatrix} \dfrac{\partial H_\beta}{\partial l_x} & \dfrac{\partial H_\beta}{\partial l_y} \\[2ex] \dfrac{\partial H_r}{\partial l_x} & \dfrac{\partial H_r}{\partial l_y} \end{bmatrix}$$

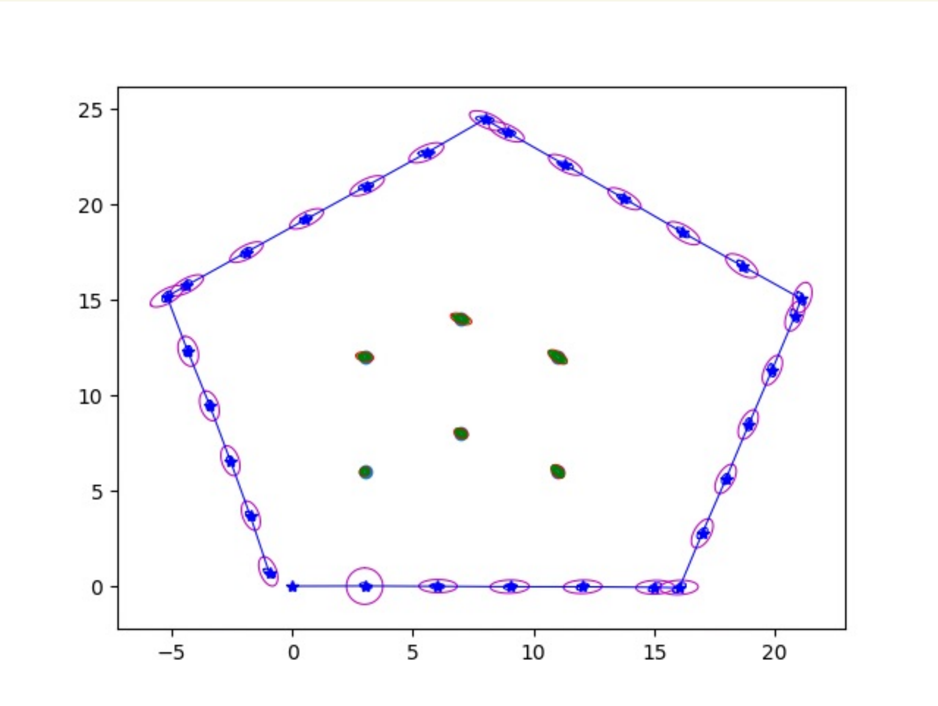
$$= \begin{bmatrix} \dfrac{P_{ty} - l_y}{(l_y - P_{ty})^2 + (l_x - P_{tx})^2} & \dfrac{P_{tx} - P_{ty}}{(l_y - P_{ty}^2) + (l_x - P_{tx})^2} \\[3ex] \dfrac{l_x - P_{tx}}{\sqrt{(l_x - P_{tx})^2 + (l_y - P_{ty})^2}} & \dfrac{l_y - P_{ty}}{\sqrt{(l_y - P_{ty})^2 + (l_x - P_{tx})^2}} \end{bmatrix}$$

Because we assume that the landmarks are independant from each other. Therefore the measurement jacobian is only relevant to its landmark.

Part 2

1. the number of landmark is 6,

2.



3.

In prediction step, the uncertainties of the poses and landmarks keep increasing. In the update step, the uncertainties decrease by the kalman gain. Because the blue circles are smaller than the pink circles. The kalman gain also regulates the weights of previous estimate and current measurment to minimize the error.

4.

The ground truth positions of the landmarks are within the ellipse. The estimated locations contain the true pos which means that the model is accurate.

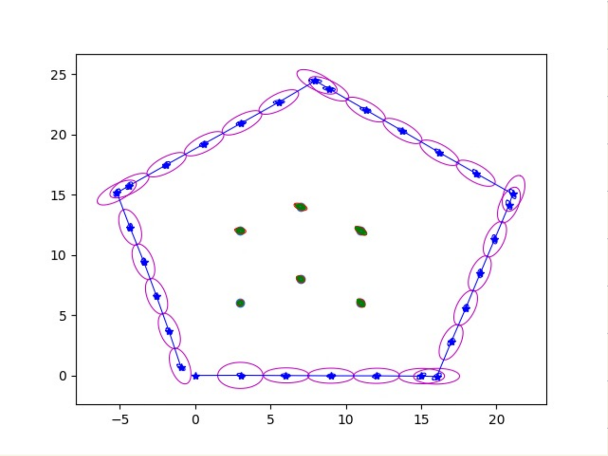
```
the Euclidean distance of landmark 1 is 0.010655310843664546
the Mahalanobis distance of landmark 1 is 0.0003920214508974523
the Euclidean distance of landmark 2 is 0.012632101500087468
the Mahalanobis distance of landmark 2 is 0.00047411524269693096
the Euclidean distance of landmark 3 is 0.002144871808856005
the Mahalanobis distance of landmark 3 is 9.251090140745066e-05
the Euclidean distance of landmark 4 is 0.009215575639476834
the Mahalanobis distance of landmark 4 is 0.0003459755209737539
the Euclidean distance of landmark 5 is 0.002381356720203794
the Mahalanobis distance of landmark 5 is 0.00010585742504228388
the Euclidean distance of landmark 6 is 0.006028633603156945
the Mahalanobis distance of landmark 6 is 0.000250488704080797
```

The Euclidean distance is the length between the true points and estimated place. The values are small Indicate the model is accurate The Mahalanobis distance is the length between a point and a distribution. The small Mahalanobis value also indicates that our model is accurate

Part 3

1. As we update the covariance matrix, the initial zero terms became non-zero terms. This is caused by the H matrix in the update function where we change P matrix by the measurement and control input. We assumed that the landmarks are independant on each other but in reality they are dependant.

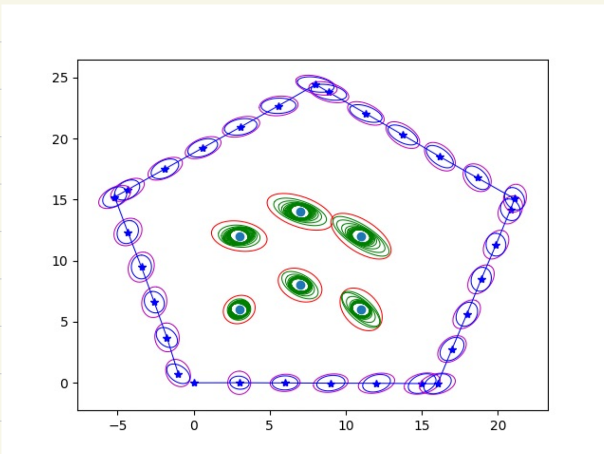
2. ① change σ_x, σ_y, σ_α $*2$



$\sigma_x = 0.5$ $\sigma_y = 0.2$ $\sigma_\alpha = 0.2$

changes in $\sigma_x, \sigma_y, \sigma_\theta$ result in making the prediction of the robot pose bigger. The reason is that it only affects the R_t term in the prediction step, thus only the uncertainty of the pose is increased.

② change in σ_β, σ_r $*\ 5$



$\sigma_\beta = 0.05$ $\sigma_r = 0.4$

changes in σ_β, σ_r results in bigger uncertainty of the pose and the landmarks. Because in the update step, the measurement covariance influence the Kalman gain. Resulting in bigger uncertainty of all states including the pose and the landmarks.

3.

① Based on the Kalman gains, we can select constant N landmarks that influence the most to our EKF algorithm.

② Instead of a for loop, we can vectorize the calculation to shorten the time for iterations.

③ Because the robot can only observe the landmarks within its range. We can discard landmarks which are out of the robot's sight.