

**Problem 1**

---

**1.1**

---

variables for each particle are:  $x, y, \theta$ .

**1.2**

---

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \\ \theta_{t+1} \end{bmatrix} = \begin{bmatrix} x_t \\ y_t \\ \theta_t \end{bmatrix} + \begin{bmatrix} \frac{r}{2}(v_l + v_r) \cos(\theta_t) \Delta t \\ \frac{r}{2}(v_l + v_r) \sin(\theta_t) \Delta t \\ \frac{r}{2}(v_r - v_l) \Delta t \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix}$$

**1.3**

---

$$\begin{bmatrix} x_{t+1}^i \\ y_{t+1}^i \\ \theta_{t+1}^i \end{bmatrix} = \begin{bmatrix} x_t^i \\ y_t^i \\ \theta_t^i \end{bmatrix} + \begin{bmatrix} \frac{r}{2}(v_l + v_r) \cos(\theta_t^i) \Delta t \\ \frac{r}{2}(v_l + v_r) \sin(\theta_t^i) \Delta t \\ \frac{r}{2}(v_r - v_l) \Delta t \end{bmatrix} + \begin{bmatrix} \delta_X^i \\ \delta_Y^i \\ \delta_\theta^i \end{bmatrix}$$

where  $\delta_X^i, \delta_Y^i, \delta_\theta^i$  are the noise.

**1.4**

---

As there are only robots in the field, there is no dynamic obstacle.

$$p_{hit}(z|x, m) = \eta \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2} \frac{(z - z_{exp})^2}{\sigma^2}\right)$$

$$p_{unexp} = 0$$

$$p_{rand} = \eta \frac{1}{z_{max}}$$

$$p_{max} = \eta \frac{1}{z_{small}}$$

$$p(z|x, m) = \alpha_{hit} \cdot p_{hit} + \alpha_{unexp} \cdot 0 + \alpha_{rand} \cdot p_{rand} + \alpha_{max} \cdot p_{max}$$

**1.5**

---

The presence of other robots does not affect the sensor model of the landmark sensors since robots do not have sensors. The sensor model for the landmark sensors remains the same as described in problem 1.4.

**1.6**

---

If they know each other's positions, we will not have dynamic obstacles. If they don't know each other's positions, we will have uncertainty.

**1.7**

---

The correction phase for the sensor model in the particle filter can be implemented as follows:

For each particle  $i$ : 1. Initialize the weight  $w_t^i$  to 1.

2. For each sensor  $j$ : a. Compute the expected measurement  $z_{exp}^j$  based on the particle's position and the map. b. Compute the likelihood  $p_{hit}^j$  using the formula:

$$p_{hit}^j(z^j|x_t^i, m) = \eta \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left(-\frac{1}{2} \frac{(z^j - z_{exp}^j)^2}{\sigma_j^2}\right)$$

c. Compute the weight update  $w_t^i$  for sensor  $j$  as:

$$w_t^i = w_t^i \cdot p_{hit}^j$$

3. Normalize the weights of all particles:

$$w_t^i = \frac{w_t^i}{\sum_{i=1}^N w_t^i}$$

4. Resample the particles based on their weights.

Note: In the above equations,  $z^j$  represents the actual measurement from sensor  $j$ ,  $\sigma_j$  is the standard deviation of the sensor's measurement noise, and  $\eta$  is a normalization constant.

This correction phase accounts for the sensor measurements and updates the weights of the particles accordingly, allowing the particle filter to converge towards the true robot positions.

#### 1.8

---

In this setup, the correspondence problem can occur. Sensors cannot associate their data with specific robots without knowing which one is in front of them.

#### 1.9

---

More correspondence would occur as the robots also don't know which sensor data is which. This would lead to more uncertainty in the estimation. There are 24 combinations of correspondence for 3 robots; we assign 24 uniform probabilities for the the sensor data and by observations, we can update the probabilities and gain insight about which sensor data belongs to which.

#### 1.10

---

The initial points in Kalman filter are usually known but in this case, it's not. We should start with some random points and correct them and wait until  $p(z|x)$  converges; we can get rid of minimal  $p(z|x)$  observations. The problem with this method is the cost of processing.