Exercise Sheet 4

Topic: EKF-SLAM

Exercise 4.1: EKF-SLAM

In this exercise, you will implement an EKF-SLAM algorithm. We assume the robot moves in the 2D plane, for example, a wheeled robot with differential drive that moves on the floor inside a building. This means the robot state $\boldsymbol{\xi}_t = (x_t,\ y_t,\ \theta_t)^T$ is 3-dimensional and composed of the 2-dimensional position x_t, y_t in the plane and the robot heading θ_t. We model the robot motion with an odometry-based motion model in this exercise, i.e. the state-transition model is

$$\boldsymbol{\xi}_t = g(\boldsymbol{\xi}_{t-1}, \boldsymbol{u}_t) + \boldsymbol{\epsilon}_t := \boldsymbol{\xi}_{t-1} + \begin{pmatrix} u_{tr}\cos(\theta_{t-1} + u_{r1}) \\ u_{tr}\sin(\theta_{t-1} + u_{r1}) \\ u_{r1} + u_{r2} \end{pmatrix} + \boldsymbol{\epsilon}_t, \qquad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{dt})$$

The action $\mathbf{u}_t = (u_{tr},\ u_{r1},\ u_{r2})^T$ is given by translational (u_{tr}) and rotational (u_{r1}, u_{r2}) motion measurements obtained from wheel odometry. For the noise covariance of the state-transitions, we assume

$$\boldsymbol{\Sigma}_{dt} = \begin{pmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.01 \end{pmatrix}$$

The robot measures the range r and bearing ϕ to 2D landmark points $\mathbf{l}_j = \left(l_{j,x},\ l_{j,y}\right)^T$ in the environment in the horizontal plane. The full state vector $\mathbf{x}$ of the EKF is the vector stacked from robot state $\boldsymbol{\xi}_t$ and all landmark points $\mathbf{l}_j$. The robot measures multiple landmarks in a time step for which we assume the association $c_{t,i} = j$ of measurements $\mathbf{z}_{t,i} = \left(r_{t,i},\ \phi_{t,i}\right)^T$ to landmarks j known. The observation model is

$$\mathbf{z}_{t,i} = h\left(\mathbf{x}_t, c_{t,i}\right) + \boldsymbol{\delta}_{t,i} := \begin{pmatrix} \left\| (x_t, y_t)^T - \left(l_{j,x}, l_{j,y}\right)^T \right\|_2 \\ \operatorname{atan2}\left(l_{j,y} - y_t, l_{j,x} - x_t\right) - \theta_t \end{pmatrix} + \boldsymbol{\delta}_{t,i}, \qquad \boldsymbol{\delta}_{t,i} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{mt,i})$$

For the observation noise of an individual landmark measurement, we assume

$$\boldsymbol{\Sigma}_{mt,i} = \begin{pmatrix} 0.1 & 0 \\ 0 & 0.1 \end{pmatrix}$$

The complete observation model in each time step, $\mathbf{z}_t = h(\mathbf{x}_t) + \boldsymbol{\delta}_t$ with $\boldsymbol{\delta}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{mt})$ stacks the M measurements in a single vector $\mathbf{z}_t = \left(\mathbf{z}_{t,0}^T, \cdots, \mathbf{z}_{t,M-1}^T\right)^T$. Analogously, we write $h(\mathbf{x}_t) := \left(h\left(\mathbf{x}_t, x_{t,0}\right)^T, \cdots, h\left(\mathbf{x}_t, x_{t,M-1}\right)^T\right)^T$. The covariance $\boldsymbol{\Sigma}_{mt}$ is formed from the individual measurement covariances,

$$\boldsymbol{\Sigma}_{mt} = \begin{pmatrix} \boldsymbol{\Sigma}_{mt,0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{mt,1} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{\Sigma}_{mt,M-1} \end{pmatrix}$$

a) Determine the analytic Jacobians of the state-transition function $g(\mathbf{x}_{t-1}, \mathbf{u}_t)$ and the observation function $h(\mathbf{x}_t)$ for the robot pose $\boldsymbol{\xi}$ and landmark positions $\mathbf{l}_j$.

b) Obtain the code sample and data for this part of the exercise from the course webpage. The archive contains three folders: data, matlab, plots. Implement EKF prediction and correction to localize the robot and map the landmarks by finalizing the code in the files **`prediction_step.m`** and **`correction_step.m`**.

c) Run your code and visualize the robot pose and landmark position estimates using the provided plotting functions.