University of Technology Nuremberg
Lecture: Introduction to Mobile Robotics
Held at the Technical University of Munich
Winter term 2022/23

Department of Engineering

Prof. Dr. Wolfram Burgard
Michael Krawez
Reihaneh Mirjalili

Sheet 7

Topic: Particle Filter

Exercise 1: Particle Filter

In the following you will implement a complete particle filter. A code skeleton with the particle filter work flow is provided for you. A visualization of the particle filter state is also provided by the framework.

The following folders are contained the pf_framework.zip:

data This folder contains files representing the world definition and sensor readings used by the filter.

code This folder contains the particle filter framework with stubs for you to complete.

You can run the particle filter in the terminal: `python particle_filter.py`. It will only work properly once you filled in the blanks in the code.

(a) Complete the code blank in the `sample_motion_model` function by implementing the odometry motion model and sampling from it. The function samples new particle positions based on the old positions, the odometry measurements δ_{rot1}, δ_{trans} and δ_{rot2} and the motion noise. The motion noise parameters are:

$[\alpha_1, \alpha_2, \alpha_3, \alpha_4] = [0.1, 0.1, 0.05, 0.05]$

The function returns the new set of parameters, after the motion update.

(b) Complete the function `eval_sensor_model`. This function implements the measurement update step of a particle filter, using a *range-only* sensor. It takes as input landmarks positions and landmark observations. It returns a list of weights for the particle set. Instead of computing a probability, it is sufficient to compute the likelihood $p(z|x, l)$. The standard deviation of the Gaussian zero-mean measurement noise is $\sigma_r = 0.2$.

(c) Complete the function `resample_particles` by implementing stochastic universal sampling. The function takes as an input a set of particles and the corresponding weights, and returns a sampled set of particles.