Department of Computer Science **FAST-NUCES** **Lahore Campus**

Robotics Technology

First Mini Project Fall 2024
Assigned on 11/11/2024 Deadline 24/11/2024 before 11:59 p.m.
Submission: GCR Weight 5%

Instructions:

Please remember that PLAGIARISM is INTOLERABLE and anyone found involved in it will get -4 marks (i.e. 100% penalty) in this assignment.

A TOUR OF THE ROBOT

Primary Objective of assignment:

Implement and compare the performance of a Histogram filter and that of a Particle filter for a robot navigating through a 2D environment represented as a 2D-grid of discrete cells. This will help us understand probabilistic state estimation and robot localization in a known environment with obstacles.

Introduction

We are going to create a simulation of a robot that uses a grid-based representation of a 2D floor to track and manage motion of a robot that will return to the starting position after visiting several pre-defined positions in order on the given floor. The robot will receive noisy sensor measurements and motion commands during the motion, which it will use to update its belief about its location to manage the movement plan goal.

Part a) [Define and Load Environment]

i) In this part we are going to manually define a map representing a floor plan with areas marked as obstacles (Walls, Closed Areas without an entry, object permanently placed in the environment etc.) and free cells. Every cell of the map will also contain estimated information about the nearest static obstacle along the horizontal and vertical directions. This information will be saved in a CSV file with pre-defined format.

ii) Create a Python class that load a map in memory as a multidimensional array and provide interface to access the map information.

Part b) [Simulate Range Sensor]

i) Assume the robot has a range sensor that detects the distance to the nearest obstacle in all four directions (up, down, left, right) from the present position of the robot. In this part we will simulate this sensor by getting the actual estimated distances from the Map and use random noise to create the noisy sensor readings that the robot will receive.

Part c) [Basic Implementation]

i) [**Initialization**] Load an approximate initial position and orientation of the robot and the goal cell(s) that must be visited by the robot before returning to the starting position. Create initial belief of the robot using 2D-Gaussian to assign initial belief to all cells in the map. For this we will create a class that manages this belief and provide additional functionality of initializing belief using other distributions like Uniform etc.

Robotics Technology

First Mini Project Fall 2024
Assigned on 11/11/2024 Deadline 24/11/2024 before 11:59 p.m.
Submission: GCR Weight 5%

ii) **[Motion Model Simulation]** Next we are going to implement the motion model of the robot. The motion model will be noisy and hence will be probabilistic as well. For each action (i.e. **T**urn and **M**ove), propagate the belief probabilities according to the motion model. The turn will rotate the robot in counter clock-wise direction by a fixed approximate angle (e.g. 1^o) and move will move it in the forward direction (i.e. direction in which head points) by a fixed distance (e.g. 1 cm) and then stop. Both the turn and move commands will be noisy and hence we will use a probabilistic model to simulate this model.

Part D) [Histogram Filter Implementation]

i) [Implement a single command]

a. Using the motion model of the robot take a simple action (e.g. Turn, Move) and hence update the belief after the action.
b. Further update the belief after taking sensor readings
c. Normalize the belief

ii) [Control Robot Movement] The path planning will be a very simple loop implementing a greedy approach. We will estimate the robots present position and then try to go to a free neighboring cell that is nearest to the next goal cell in the tour plan. To move we will generate commands (sequence of turns and move) to move in the desired direction till we reach back after (approximately) visiting each location in the desired plan.

Part D) [Test and Create Report]

Test your implementation by generating multiple scenarios and viewing the position of robot as it moves.

REPEAT THE ABOVE STEPS USING PARTICLE FILTERS INSTEAD OF HISTOGRAM FILTERS

Part D) [Submit a detailed Report]

i) Implementation Details:

a. Describe how each part of the filter (prediction, update, normalization) was implemented.
b. Explain the motion model and how noise was incorporated in sensor readings and movements.

ii) Results:

a. Include screenshots showing belief updates over time and the robot's path from start to goal.
b. Discuss the impact of sensor and motion noise on the localization and path planning accuracy.
c. Compare the performance of two filter in different environments and identify limitations observed.