Intelligent Robotics: Practice Problems

December 4, 2018

- ***Question 1:***

  You are developing the localization module for the robot soccer domain. The robot is provided a map of the soccer field, along with the six color-coded markers and the two goals along the boundary. You decide to explore a Particle Filter (PF) algorithm for keeping track of the robot's location over time.

  1. Describe the input(s) and output(s) of the PF algorithm.
  2. Describe the steps of the PF algorithm and the corresponding mathematical equations.
  3. Assume that you have three particles with importance weights $\langle 0.2, 0.6, 0.2 \rangle$, and now want to create 5 particles through resampling based on the algorithm discussed in class. Show the working of this algorithm.

- ***Question 2:***

  You are developing the software system for an autonomous car, and one of your tasks is to enable the car to simultaneously build/revise the map of the area and localize itself in the map. You decide to explore the use of a Particle Filter-based Simultaneous Localization and Mapping algorithm for this problem. You are provided the following additional information:

  – The map of the area is discretized into 2D grids of reasonable size.

  – The car uses a laser range finder as the primary source of information.

  – From a given position on a road, the car can move forward or backward, or turn clockwise or anti-clockwise by 90^o.

  1. Assuming you are using the FastSLAM algorithm, describe its input(s) and output(s).
  2. Describe the *states*, *actions* and *observations*, and their representation, in this algorithm.
  3. Describe the key mathematical steps in the FastSLAM algorithm.

- ***Question 3:***

  You are developing the software for a robot soccer team. One of your tasks is to develop a software module that will enable a robot to track the position of the ball on the soccer field. You decide to explore the use of a Kalman Filter (KF) for this problem. You are given the following additional information:

  – The ball's position $\mathbf{x}$ is given by its (x, y) coordinates relative to the robot.

  – The basic equations of Physics hold. For instance:

  $$\mathbf{x}_t = \mathbf{x}_{t-1} + \mathbf{u}_{t-1}\Delta t$$

  where $\mathbf{u} = (u_x, u_y)$ is the ball's velocity.

  – You have a working vision system that can detect the ball in an image. You determine in an initial study that the ball's relative distance is over-estimated—the measured position is 1.2 times the actual position.

  1. Define the *states*, *controls/actions* and *observations* for this problem.
  2. Provide the state update equation and measurement equation for this problem in terms of the state, control and observation. Populate the corresponding matrices based on the information given.
  3. The KF is an instance of a Bayesian filter. Provide a mathematical description of the two fundamental steps in such a filter.

- ***Question 4:***

  Consider a robot answering questions about objects in a scene based on information extracted from images of the scene. You have the following additional information:

  – Each object is characterized by *size* and *shape*, which can take values from $\{small, medium, large\}$ and $\{circle, triangle, rectangle\}$ respectively.

  – Each image has been segmented to provide regions of interest (ROIs). Each ROI can be processed using two *knowledge-producing* actions *size* and *shape*. These actions output the ROI's size and shape as one of $\{small, medium, large\}$ and $\{circle, triangle, rectangle\}$ respectively, but do not always output the correct value.

  1. Consider a robot analyzing a ROI to answer the question *"is there a small rectangle in the ROI?"* You are asked to formulate this task as a partially observable Markov decision process (POMDP). What are the components of a POMDP, and how does it differ from an MDP? Why does it make sense to formulate this task as a POMDP?

  2. Describe the components of the POMDP in the *Cassandra* format. You are given the following additional information:

     – States include all possible combinations of the *size* and *shape* attributes, and a *terminal state absb* that is reached by executing a *terminal action* (see below).

     – Actions include the knowledge-producing actions, and terminal actions (*yes* and *no*) that lead to the terminal state.

     – Observations include possible outcomes of all the non-terminal actions.

     – The state transition function is a square matrix for each action, with the number of rows equal to the number of states.

     – The observation function is a matrix for each action, with the rows representing states and the columns representing observations. The probability of an observation matching the true value of the corresponding attribute (*size* or *shape*) is 0.9—the remaining probability is distributed over other outcomes.

- For the reward specification, the *shape* action is twice as computationally expensive as the *size* action. For terminal actions, a large positive (negative) reward is obtained for providing the correct (incorrect) answer.

- ***Question 5:***

  Consider a robot navigating in a 3×4 grid world shown in Figure 1. Your objective is to provide the robot a policy that can be used to move from any cell to the *"goal"*. You are given the following information:

  – The robot cannot move to any grid cell with an obstacle (shaded black in Figure 1). All other valid grid cells are numbered, starting from "0" in the top left corner, scanning one row at a time (left to right) until the bottom right corner is reached.

  – Actions include $\{north, south, west, east, stay\_put\}$.

  – Any action that attempts to move the robot out of the grid, or to an invalid cell, causes no change in the robot's location. The action *stay_put* has the same effect. With probability 0.85, all other movement actions succeed.

  – After executing an action, the robot's location is known. Once the robot reaches the goal state, it stays there until it is reset.

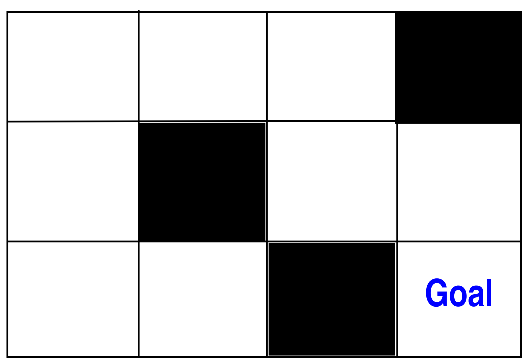

Figure 1: A 3×4 grid world; shaded cells represent zones with obstacles.

Formulate the navigation problem as a Markov decision process (MDP) by describing the components of the MDP in the *Cassandra* format. Show the state transition function for at least two non-terminal actions and one terminal action. Your reward specification should assign a cost for each action and a large positive reward for reaching the goal. above.