Task Sheet 3

Advanced Autonomous Robotics

Deadline 10:00am May 10, 2023

Review on May 14 & 15, 2023

Lecture: *Advanced Autonomous Robotics*, Summer Term 2024

Lecturer: Prof. Dr.-Ing. Heiko Hamann

Tutor: Jonas Kuckling & Paolo Leopardi

Task 3.1 Theoretical questions

a) Explain the difference between uncertainty and noise. (*2 points*)

b) What is the difference between sensing and perception? How does a world model relate to these ideas? (*4 points*)

c) What does it mean to treat perception as inference? (*2 points*)

d) What are the reasons to use ROS 2 for robotics? (*2 points*)

Task 3.2 A robot that acts to minimize uncertainty (*implementation*)

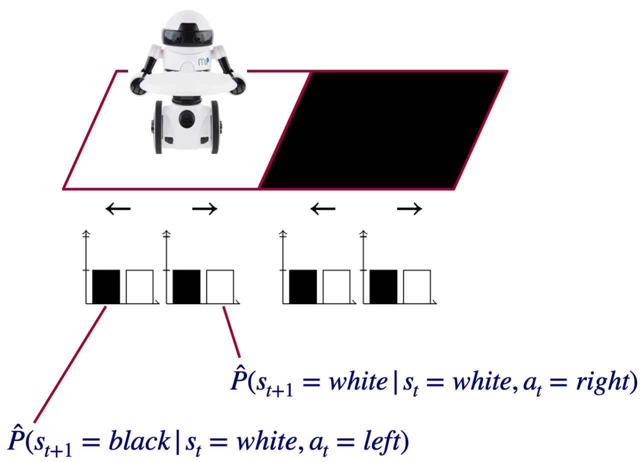

Figure 1: The uncertainty-minimizing robot's little world.

We want to implement the *uncertainty-minimizing robot* from the introduction chapter of the course (see Fig. 1).

The environment in which the robot operates is relatively simple, consisting of only two tiles: a white tile on the left and a black tile on the right. The robot can sense the color of the tile $s \in \{\text{white}, \text{black}\}$ it is currently on and can choose to move left or right ($A = \{\text{left}, \text{right}\}$). If the robot is placed on the black tile and chooses to go right despite the world ending there, it will just stay stopped (similarly for choosing to go left on the white tile).

Initially, the robot has no information about its environment, and it must learn from experience. To achieve this, we provide the robot with a data structure that maintains a histogram H of the frequency of occurrences of its past experiences, including its previous position s_t, its chosen action a_t, and the color of the tile s_{t+1} it sensed next.

We interpret the entries of the histogram as conditional probabilities:

$$P(s_{t+1} = \text{white} | s_t = \text{black}, a_t = \text{right}) = \frac{H(s_{t+1} = \text{white}, s_t = \text{black}, a_t = \text{right})}{H(s_{t+1} = \text{white}, s_t = \text{black}, a_t = \text{right}) + H(s_{t+1} = \text{black}, s_t = \text{black}, a_t = \text{right})}. \tag{1}$$

The only strategy that we make the robot to follow is to always go for the maximal probability P. The robot chooses action $a_{t+1} = \arg\max_{a_t} P(s_{t+1}|s_t, a_t)$.

a) Implement a tiny simulator of the world as given in Fig. 1.

b) Implement the robot's histogram H and its policy $a_{t+1} = \arg\max_{a_t} P(s_{t+1}|s_t, a_t)$. Make sure you choose the correct tie breaker: if $\max_{a_t} \hat{P}(s_{t+1}|s_t, a_t) = 0.5$), the robot makes a random choice by choosing an action uniformly randomly.

c) Let the robot run several times for several steps. Can you reproduce the three behaviors as discussed in the lecture? Choose an appropriate way of presenting your resulting robot behaviors in your submission.

Submission instructions:

For your submission, please submit the answers to the theoretical questions (as a machine typed PDF file), the code (and some short documentation on how to execute it), as well as a presentation of your results. The presentation of your results is up to you and can take many forms. For example, you could submit a short PDF file with plots or screenshots from your terminal. Alternatively, you could submit a video of the screen capture, where you explain what is happening. The spirit should be to not just complete the programming task and done - but to play with your little sim a bit. Explore for yourself. Let us know if you found something interesting. Have fun!