VI51_2013P_FS01_01

utbm
université de technologie
Belfort-Montbéliard

VI51: Virtual Life Simulation - Final Exam P2013

Duration: 2h.
No document nor calculator nor smart phone nor touchpad allowed.
Each part should be written on a separate sheet.
English recommended, French accepted.

Part 1: Markov models, The breakfast's paradox (by F. Gechter, 4 points)

Everybody has encountered the breakfast's paradox, which may transform a great sunny day into a deep journey into nervousness. Indeed, when a piece of bread with butter on one side falls from the table, it hits the ground on the butter side in most cases. In order to model this phenomenon which can trigger huge nervous breakdowns, we decided to build up an experiment. We prepare two desks which are covered by a lot of bread pieces. The right desk corresponds to blank pieces (i.e. without butter) whereas the left desk corresponds to bread pieces with one side covered by butter. At each time step, one piece is chosen among the pieces on the two desks without knowing which desk is chosen. The result obtained is denoted B if the pieces hits the ground on the butter side and OK if the side that hits the ground has no butter. For the left side desk, we consider that the probability of hitting the ground with the butter side is equal to 70%. We also have a probability of 50% of changing desk at each time step.

Question 1.1:

Draw the breakfast's paradox Markov graph. Which Markov model suits to this example? Why?

Question 1.2:

Let consider the following observation: B OK OK B OK B. Which algorithm should be used to compute the most probable sequence of desks changing that leads to this observation? Compute it.

Part 2: Cleaner robot (by F. Lauri, 6 points)

An agent wants to gather as many cans as possible in an area that can be represented by the following graph $G = (N, E)$:

- the set of nodes is $N = \{A, B, C\}$
- the set of edges is $E = \{\{A, B\}, \{B, C\}\}$

Prior to any learning episode, **1 can** is located on nodes A and **3 cans** are on node B. On each node, the agent:

- can sense how many cans remains there,
- can pick up one of the remaining can.
- can go to one of the adjacent node.

Node C represents a dangerous place where the agent progression is stopped. In other words, its learning episode is stopped.

Question 2.1:

Define as precisely as possible the MDP representing this task.

Question 2.2:

Recall the algorithm Q-Learning.

Question 2.3:

Assuming the agent always starts from node A, apply 3 episodes of the Q-Learning, indicating all the parameters and choices used by the algorithm (starting states, transitions, values of γ and α...) and show the final matrix of Q-values.

Question 2.4:

What is the optimal policy for this task?

Part 3: Environment Model and Movement Behaviors (by S. Galland, 10 points)

Question 3.1:

Explain the principle and the main advantage of the influence/reaction approach.

Question 3.2:

What is the difference between the "Flee" kinematic behavior and the "Flee" steering behavior?

Question 3.3:

Provide the algorithm of the **combined steering behaviour** "Evade".

Exercise 3.4:

Assuming a 2D continuous environment, you want to write the behaviours of several mobile entities that are moving inside a room because they are trying to reach a point. You should avoid collision among them. Write a **force-based algorithm** that permits to avoid collision, and reaching a goal point in the same time.

Exercise 3.5:

You want to write a perception algorithm for agents in a 2D environment, which supports occlusion culling (occluded objects must not be seen by the agent).
Assuming that frustum culling is done in by the function: `List<Object> frustumCulling()`, and this function is already provided, propose an occlusion culling algorithm.
If you have made several design hypothesis, you must quickly explain them.