CS 5100: Foundations of Artificial Intelligence (Fall 2023) Chris Amato
Student name: Bharat Chawla (Due) Friday, September 22rd
Colloborators:

Introduction, Agents, Search

1 Introduction

1. To what extent are the following computer systems instances of artificial intelligence:

   (a) Supermarket bar code scanners

   Supermarket bar code scanners is a simple system designed to read and display the code. Hence, it is not an instance of artificial intelligence.

   (b) Web search engines

   It is an instance of AI as it is continuously learning the problem of determining the applicability of a web page to a user query. Additionally, it determines how content gets ranked to based of end user's needs. However, retrieval of billions of pages in seconds is tied to indexing problem in a database design, not in artificial intelligence.

2 Intelligent Agents

A. For each of the following assertions, say whether it is true or false and support your answer with examples or counterexamples where appropriate.

(a) It is possible for a given agent to be perfectly rational in two distinct task environments.

True. Let's consider two distinct environments based on betting the outcomes of rolling a dice. In one environment, the dice are fair and in the other, they are biased to give 3. The agent can bet on what the outcome of the dice will be, and are rewarded for guessing the outcomes correctly. The agent that always bet on 3 will be rational in both scenarios.

(b) Suppose an agent selects its action uniformly at random from the set of possible actions. There exists a deterministic task environment in which this agent is rational.

True. For instance, an environment where all the set of possible actions receive the same reward, it doesn't matter which action you take, and selecting them randomly is rational.

(c) There exists a task environment in which every agent is rational.

True. For example, in a single state environment such that all actions lead to same reward, regardless of which action is taken will satisfy the stated property.

(d) A perfectly rational Go agent always wins the game.

False. Unless the agent has access to unlimited computational resources to analyze all possible and counter-moves to ensure it wins the game. Furthermore, the game is highly strategic and tactic based, and agent's victory is not solely dependent on their rationality, but also on adaptability and strategy.

B. Properties of task environments

- Fully observable vs. partially observable
- Single agent vs. multiagent
- Deterministic vs. stochastic
- Static vs. dynamic
- Discrete vs. continuous

For each of the following activities, characterize the task environment it in terms of the properties listed above.

(a) Assembling an IKEA Chair.

Fully observable, Single agent, Deterministic, Dynamic, Discrete

(b) Playing Atari games.

Fully observable, Single agent, Stochastic, Dynamic, Discrete

(c) A boxing match.

Fully observable, Multi-agent, Deterministic, Dynamic, Continuous

(d) Driving a car.

Fully observable, Single agent, Stochastic, Dynamic, Continuous

(e) Search and rescue on a mountain.

Partially observable, Multi-agent, Stochastic, Dynamic, Continuous

3 Search

Your goal is to navigate a robot out of a maze. The robot starts in the center of the maze facing north. You can turn the robot to face north, east, south, or west. You can direct the robot to move forward a certain distance, although it will stop before hitting a wall.

(a) Formulate this problem. How large is the state space?

State: Each state includes a robot's position (x,y) and it's orientation, where x and y are the coordinates.
Initial state: Initial state is the starting configuration of the robot i.e. (x_initial, y_initial) and facing north, where x_initial and y_initial represents the center of the maze.
Actions: In this environment, each state has two actions: first, turn the robot to face north, east, south, or west and second, direct the robot to move forward a certain distance in the orientation it is currently facing until it runs into a wall or maze's boundary.
Transition model: The actions have their expected effects except turning or moving forward a certain distance in the orientation it is currently moving won't move through walls or maze's boundary.
Goal test: This checks whether the robot has successfully navigated out of a maze i.e. where the robot's position is outside or at the maze's boundary.
Path cost: Each step costs 1, so the path cost is the number of steps taken by robot to navigate out of a maze.

The state space will depend on the dimensions of the maze. For our simplicity, we can define the dimensions of maze as r * c cells. In the given problem, the state space is determined by number of positions (x,y) within the maze and number of possible orientation that a robot can face (4 directions). Therefore, state space is r * c * 4.

(b) From each point in the maze, we can move in any of the four directions until we reach a turning point, and this is the only action we need to do. Reformulate the problem using these actions. Do we need to keep track of the robot's orientation now?

No, we don't need to keep track of the robot's orientation now if the robot is allowed to move in any of the four directions until it reaches a turning point, and this is the only action it is allowed to perform.

State: Each state includes a robot's position (x,y), where x and y are the coordinates.
Initial state: Initial state is the starting configuration of the robot i.e. (x_initial, y_initial), where x_initial and y_initial represents the center of the maze.
Actions: In this environment, robot to move forward a certain distance in the orientation it is currently facing until it reaches a turning point.
Transition model: The actions have their expected effects i.e. if a robot chooses to move in a certain direction, it's position (x,y) will change accordingly, but it's orientation won't until it reached a turning point.
Goal test: This checks whether the robot has successfully navigated out of a maze i.e. where the robot's position is outside or at the maze's boundary.
Path cost: Each step costs 1, so the path cost is the number of steps taken by robot to navigate out of a maze.

The state space will depend on the dimensions of the maze. For our simplicity, we can define the dimensions of maze as r * c cells. In the given problem, the state space is determined by number of positions (x,y) within the maze and have simplified the state space by eliminating to consider for robot's possible orientation. Therefore, state space is r * c.

(c) In our initial description of the problem we already abstracted from the real world, restricting actions and removing details. List four such simplifications we made.

Four simplifications made in the initial description are:

1. The robot can only move in one of the four given orientations.
2. Assumed it's a deterministic maze where the dimensions and obstacles are fixed and known in advance.
3. Assumed there are no resource constraints on the robot i.e. battery, sensors, or computational capacity.
4. Assumed there are no dead ends in the maze for robot to be aware of.