Fall 2022 - CSC545/645 Artificial Intelligence - Assignment 2

Due date: Thursday, September 15, 2022, 2pm. Please create a folder called assignment2 in your local working copy of the repository and place all files and folders necessary for the assignment in this folder. Once done with the assignment, add the files and folders to the repo with svn add *files,folders* and then commmit with svn ci -m "SOME USEFUL MESSAGE" *files,folders* .

Exercise 2.1 [6 points]
Read chapter 3.1 – 3.5 (problem-solving agents – uninformed search strategies) of the textbook.

1. **Explain why problem formulation must follow goal formulation.**
   [1 point]
   Problem formulation takes an abstract view of the world to determine states and how the actions an agent can perform would change those states. Since the agent would only know if its reached the goal by knowing if it is in “a goal state” (or if an action would place it in a goal state) the goal formulation must come first to be able to then describe the goal as a state for the problem formulation.

2. **The textbook says that we would not consider problems with negative path costs. In this exercise, we explore this in more depth.**
   5 points]

   (a) Suppose that actions can have arbitrarily large negative costs; explain why this possibility would force any optimal algorithm to explore the entire state space.

   If any action can have any negative cost, then that could imply that at some point in the state space there exists some action that could drop the total cost down to zero or even negative, which means it does not matter what the cost to find that state is since it would all be removed in the end. With this state always being possible, it incentivizes the algorithm to explore the entire state space looking for this magic state and action.

(b) Does it help if we insist that step costs must be greater than or equal to some negative constant c? Consider both trees and graphs.

Yes, this does help. In contrast to the above situation of being unable to know how low the cost could go therefore encouraging infinite searching, if we know the lowest cost a set of actions could be is “c times the number of actions”, this is now a known value we can compare to the costs of other actions we already have calculated. This works both in trees, although we may need to limit how many times this negative route can be taken or make sure the cost to enter is high, and graphs since graphs wont repeat the same state anyway.

(c) Suppose that there is a set of operators that form a loop, so that executing the set in some order results in no net change to the state. If all of these operators have negative cost, what does this imply about the optimal behavior for an agent in such an environment?

This would imply that the optimal behavior would be to loop through these actions infinitely in an attempt to reduce the cost as much as possible. The program would always take these steps in that order to get back to the start state at a lower cost than before, and then repeat the process.

(d) One can easily imagine operators with high negative cost, even in domains such as route finding. For example, some stretches of a road might have such beautiful scenery as to far outweigh the normal costs in terms of time and fuel. Explain, in precise terms, within the context of state-space search, why humans do not drive around scenic loops indefinitely, and explain how to define the state space and operators for route finding so that artificial agents can also avoid looping.

Even if there are two routes from A to B and the beauty of a higher-cost route makes it more desirable (most likely through a heuristic) the end goal is to still get from A to B. The cost of navigating back towards A away from B would be immensely high (aka when initiating the infinite loop) since it is in stark opposition to the end goal. Taking more time and fuel could be worth it for some distance, but not for an infinite distance because the positive-backtracking cost is even higher than the high negative cost the scenery provides.

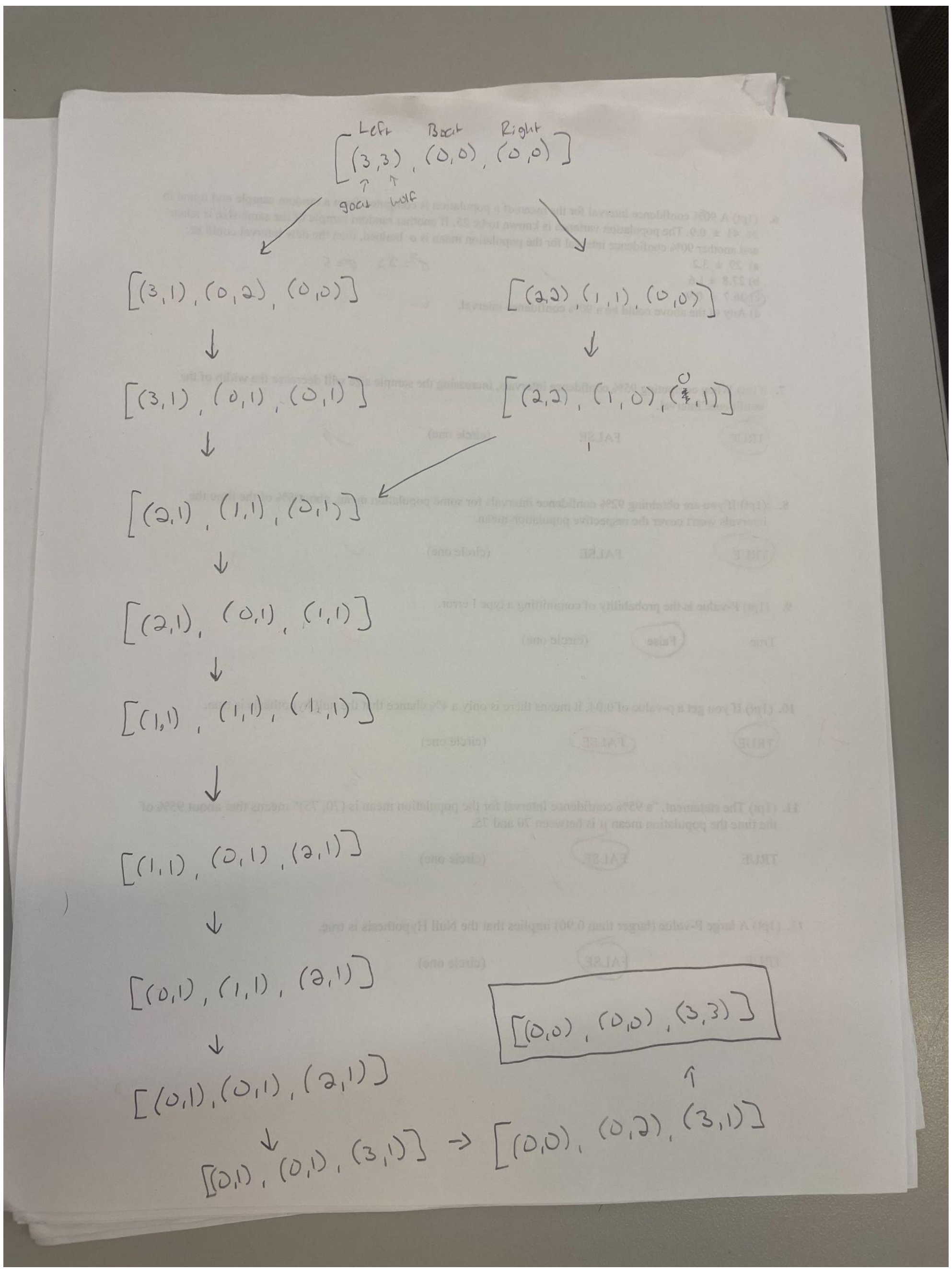
(e) Can you think of a real domain in which step costs are such as to cause looping?

Drinking coffee to immediately relieve tiredness only to crash and be more tired later - not to mention making yourself more tired more often without coffee requiring more coffee for immediate relief to only make yourself more tired, ect.

Exercise 2.2 [14 points]

The goats and wolves problem is usually stated as follows: Three goats and three wolves are on one side of a river, along with a boat that can and must hold either one or two people. Find a way to get everyone to the other side without ever leaving a group of goats in one place outnumbered by the wolves in that place. This problem is famous in AI because it was the subject of the first paper that approached problem formulation from an analytical viewpoint [Ama68].

1. **Formulate the problem precisely, making only those distinctions necessary to ensure a valid solution. Draw a diagram of the complete state space (without repeated states). [4 points]**



2. **Implement and solve the problem optimally using an appropriate search algorithm. Is it a good idea to check for repeated states?**
   [9 points]

   No it is not. For one, if we allow repeating states we would just infinitely pass one of the animals back and forth from the left side into the boat forever. Secondly, if we get past this first issue, going back into a state we have already visited would start a chain of searching we have already started and may have decided leads nowhere. There is no new information that going back into an older state could provide us.

3. **Why do you think people (humans) have a hard time solving this puzzle, given that the state space is so simple?**
   [1 point]

   People like to find literal straightforward solutions, meaning not taking steps backward or halting progress. The key to this puzzle is to constantly juggle a wolf back and forth, therefore not making “forward progress” with it when its the only thing in the boat. Although taking a step back reveals this is key to solving the puzzle, as it removes a wolf to worry about on the islands, keeping it on the boat indefinitely halts this “forward progress” mentality.

References

[Ama68] Saul Amarel. On representations and modeling in problem solving and on future directions for intelligent systems. Technical report, RCA LABS PRINCETON NJ, 1968.