Assignment No. 2 - Theoretical Assignment
AI and Robotics (236609)

Goldstein, Sharon
sharongold@campus.technion.ac.il
Computer Science
Technion - Israel Institute of Technology

Granov, Yotam
yotam.g@campus.technion.ac.il
Mechanical Engineering
Technion - Israel Institute of Technology

March 30, 2023

1 Modeling

We will provide a formal definition of the Taxi domain using the three models from assignment 1: Classical planning, Markov-Decision Process (MDP), and Partially Observable Markov Decision Process (POMDP). For simplicity, we'll assume there is no bound on the size of the grid but that some movement actions might be impossible given a taxi place.

1.1 Classical Planning - STRIPS (F, I, A, G, C):

The states include a map of the grid world, the taxi is represented by its place in the grid, and a flag that indicates if the taxi is occupied. (All of this can be saved as a total grid). The initial state is a randomly given grid where the taxi is not occupied and the taxis and passenger's places are randomly chosen. The goal states are when the passenger is in the end location.
The actions are the 6 discrete deterministic actions: move to a neighbor cell in the grid (if feasible), pick up a passenger, and drop off a passenger.
The cost C is a high positive number (20 in the taxi domain) for reaching the finite state, dropping off the passenger in the wanted place, a high negative number (-10 in the taxi domain) for unfeasible or illegal actions, like "drop-off" in an illegal place, and a low negative score (-1 in the taxi domain) for every other action.

1.2 MDP (S, A, P, R, γ):

The states S - the total grid, explained in the previous part.
The actions A - the 6 discrete actions, also explained above.
Probability $P_a(s_t, s_{t+1})$ - for a pick-up action, probability $1 - p$ to pick-up the passenger, and probability p for $s_t = s_{t+1}$ for a drop-off, probability 1 to succeed. for a movement action, if the movement is not feasible then probability 1 for $s_t = s_{t+1}$, if the movement is feasible then probability p_m for $s_t = s_{t+1}$, and $1 - p_m$ for performing the action successfully.
The reward R - the same as the cost above.

1.3 POMDP $(S, A, P, R, \gamma, \Omega, O, b_0)$:

The states S, the actions A, probability, and reward are the same as the MDP.
The observations Ω are the grids with the walls around it in the direction detected and right of that direction. The sensor function O specifies the conditional observation probabilities, returns 0.8 for the grids with a wall in the observed direction and 0.2 for the grids right of that direction. The initial belief is 1/5 for every grid where the taxi place is in the given cell or in one of its 4 adjacent cells.

2 Best First Search

In our code, we run a* with the very basic goal heuristic $h_G(n)$, which assigns a value of 1 to a node n that represents a terminal state, and 0 otherwise.

2.1 Is $h_G(n)$ admissible for the taxi domain?

$h_G(n)$ is not admissible for the taxi domain. An admissible heuristic doesn't overestimate the actual cost of reaching the goal state, so it must be less than or equal to the true cost for any state. Therefore the cost of a final state in an admissible heuristic needs to be equal to zero.

2.2 An admissible heuristic $h'_G(n)$ that dominates $h_G(n)$ for the taxi domain.

We will suggest a Manhattan distance between an occupied taxi and the passenger destinations (if there is a couple of taxis and passengers, then the minimal value). For an unoccupied taxi and an unpicked passenger, we will sum the minimal Manhattan distance between them and the minimal Manhattan distance between the passenger's location and possible destinations.
This heuristic is admissible. Because it calculates the minimal grid distance between the taxi and a drop-off, it will always return a lower estimation of the cost to reach the goal state.
This heuristic dominates the previous since it returns distance, and will always return a non-negative value, bigger than the previous heuristic.

2.3 Is the heuristic admissible for all deterministic and fully observable domains?

This heuristic is admissible for all deterministic and fully observable grid domains when the roads are parallel to the grid. In this case, it will always return a minimal distance and would not overestimate the cost. If the domain has an unparallel road, the given heuristic can return a number higher than the distance. This might cause an overestimation.

2.4 An admissible heuristic for the stochastic version of the taxi domain

In the given stochastic version of the taxi domain, the previous heuristic doesn't take into account that the initial belief is a uniform distribution over the actual cell, and the 4 adjacent cells. To count for the initial belief, we can choose the minimal Manhattan distance between the actual

place cell and the 4 adjacent cells to the given drop-off points. This will always under-estimate the excepted total cost.

3 Q Learning

In the standard version of Q-learning we saw in class how to use the GLIE epsilon-greedy exploration principle.

3.1 If and How the GLIE principles are supported in Q-learning

A general Q-learning algorithm does not have to satisfy the GLIE principles. An non-zero ensures the first property but does not ensure the second one.

The GLIE principles are supported in Q-learning when we use a greedy epsilon. In Q-learning, the agent learns an optimal action-value function by updating its estimations at every step. The agent has an exploration rate, that controls the probability to act according to a random action instead of the highest estimation. Gradually decreasing this exploration rate over time allows the agent to gather exploration at the beginning and gradually shift to exploitation as it becomes more confident and after a long time ($t \to \infty$) the policy will become greedy with respect to its learned values.
The algorithm asymptotic convergence to Q* is valid while every s, a is seen an infinite amount of times, a greedy epsilon ensures an optimal policy.

3.2 An alternative implementation of Q-learning that is GLIE

Another implementation of Q-learning which is GLIE, is a Boltzmann policy-based implementation that decreases the temperature parameter. In the Boltzmann policy implementation, the original action distribution gets divided by the temperature parameter. The agent's exploration behavior becomes a spectrum between picking the action randomly and always picking the most optimal action - the temperature parameter in the Boltzmann distribution controls the degree of exploration vs exploitation, and as the temperature decreases over time, the policy becomes increasingly greedy and satisfies the second GLIE property.