CS3243 Introduction to Artificial Intelligence

AY2021/2022 Semester 1

Tutorial 4: Revision

Important Instructions:

- There are no assignment questions for this tutorial.

Tutorial Questions

(1) Assuming that ties are broken based on alphabetical order, specify the order of the nodes that would be explored by the following algorithms. Assume graph-based implementations, and that S is the initial node while G is the goal node.

Note that you **MUST** express your answer in the form $S-B-A-F-G$ (i.e. no spaces, all uppercase letters, delimited by the dash ($-$) character), which, for example, corresponds to the exploration order of S, B, A, F, then G.

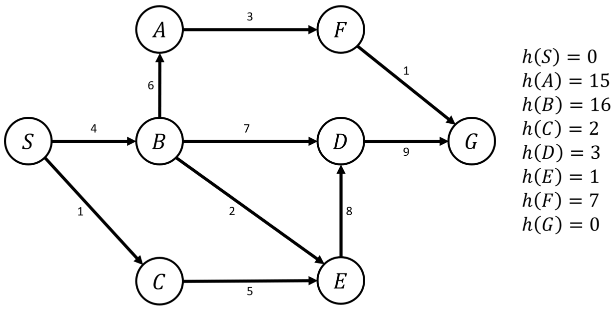

Figure 4.1. Graph for question 1.

(a) Uniform Cost Search.

(b) A^* Search.

Solution:

(a) $S-C-B-E-A-D-F-G$

(b) $S-C-E-D-B-G$

(2) Prove that the tree-based variant of the A^* **Search** algorithm is optimal when an admissible heuristic is utilized.

Solution: We want to make sure that when the goal node s_{goal} is popped, we would have found the optimal path π^*. The crucial moment is when s_{goal} is in the frontier F.

We consider the optimal path as follows:

$$\pi^* : s_0 \rightarrow s_1 \rightarrow \ldots \rightarrow s_{\text{goal}-1} \rightarrow s_{\text{goal}}$$

Because we keep optimizing the cost, we have

$$\hat{f}(s_i) \geq f_{\pi^*}(s_i) \geq f(s_i)$$

where $f_{\pi^*}(s_i)$ is the f value of node s_i along the optimal path. From the property of an **admissible heuristic**, we have:

$$\begin{aligned} h(s_i) \leq h^*(s_i) &\Longrightarrow h(s_i) \leq g_{\pi^*}(s_{\text{goal}}) - g_{\pi^*}(s_i) \\ &\Longrightarrow h(s_i) + g_{\pi^*}(s_i) \leq g_{\pi^*}(s_{\text{goal}}) + h(s_{\text{goal}}) \\ &\Longrightarrow f_{\pi^*}(s_i) \leq f_{\pi^*}(s_{\text{goal}}) \\ &\Longrightarrow f_{\pi^*}(s_i) \leq f_{\pi^*}(s_{\text{goal}}) \leq \hat{f}(s_{\text{goal}}) \end{aligned}$$

Since s_0 is the starting node, it would be the first node to get into the frontier F, and we know that $\hat{f}(s_0) = f_{\pi^*}(s_0)$. When s_0 is explored, s_1 would be added to F with $\hat{f}(s_1) = f_{\pi^*}(s_1)$; when s_1 is explored, s_2 would be added to F, and so on.

Eventually, we know that when s_{goal} is in the frontier F, there exists a node s_i along π^* which satisfies $\hat{f}(s_i) = f_{\pi^*}(s_i)$. This means that before the goal node s_{goal} is popped, examined, and returned with a non-optimal f value, another node along the optimal path must be popped first.

Then, at some point, node s_i will be popped and at that time, we have:

$$\begin{aligned} \hat{f}(s_{i+1}) &= \hat{g}(s_{i+1}) + h(s_{i+1}) \\ &= \hat{g}(s_i) + c(s_i, s_{i+1}) + h(s_{i+1}) \\ &= g_{\pi^*}(s_i) + c(s_i, s_{i+1}) + h(s_{i+1}) \\ &= g_{\pi^*}(s_{i+1}) + h(s_{i+1}) \\ &= f_{\pi^*}(s_{i+1}) \end{aligned}$$

This will repeat for $s_{i+2}, s_{i+3}, \ldots$, until s_{goal} is popped with $\hat{f}(s_{\text{goal}}) = f_{\pi^*}(s_{\text{goal}})$, which implies that

$$\hat{f}_{\text{pop}}(s_{\text{goal}}) = f_{\pi^*}(s_{\text{goal}})$$

(3) Pac-Man[1] is a maze chase video game, where the player controls a character to navigate an enclosed maze. In a typical Pac-Man game, the goal of the player is to consume all the dots placed in the maze, while avoiding the colored ghosts.

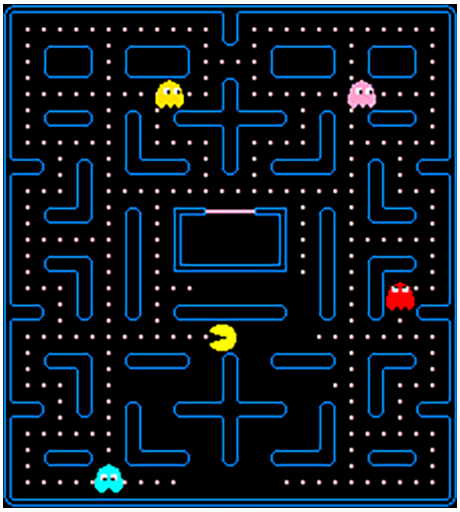

FIGURE 4.2. A typical game of Pac-Man.

However, Karen decided to create her own version of Pac-Man, which does not have any ghosts, and only has a single dot at each corner of the maze (see the figure below for a reference).

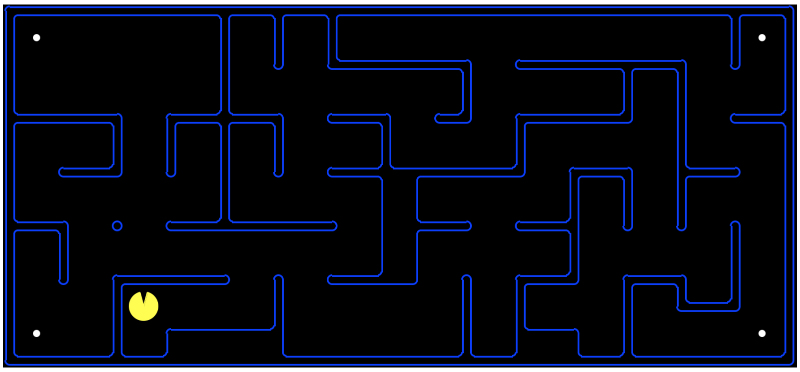

FIGURE 4.3. Karen's version of Pac-Man.

In her version of Pac-Man, the goal of the player is to consume all four dots placed in the maze as quickly as possible. Design a nontrivial, admissible heuristic so that the character, adopting an A^* **Search** algorithm with tree-based implementation, can navigate her maze optimally.

Solution: Discuss in the forum/with your project partner :)

(4) Consider the 8-puzzle problem[2]. Let h_1 and h_2 be defined as "the number of misplaced tiles" and "the sum of the Manhattan distances between current and goal positions of each tile" respectively. Suppose we define new heuristic functions h_3 and h_4, given by:

$$h_3 = \frac{h_1 + h_2}{2}$$
$$h_4 = h_1 + h_2$$

Are h_3 and h_4 admissible?

Solution: Since the Manhattan distance between the current and goal position of each tile is ≥ 1 if the tile is misplaced, hence we know that $h_1(n) \leq h_2(n)$ for all n. Furthermore, we know that $h_2(n) \leq h^*(n)$ (i.e. h_2 is admissible) since the Manhattan distance heuristic is a relaxation of the original problem. Therefore,

$$h_3(n) = \frac{h_1(n) + h_2(n)}{2} \leq \frac{h_2(n) + h_2(n)}{2} = h_2(n) \leq h^*(n)$$

where the last inequality holds since h_2 is admissible. Hence, h_3 is admissible.

On the other hand, h_4 is not admissible. Consider a board n in which moving one tile will reach the goal. In this case, $h_1(n) = h_2(n) = h^*(n) = 1$, and

$$h_4(n) = h_1(n) + h_2(n) = 1 + 1 > h^*(n)$$

[1]See Pac-Man (https://en.wikipedia.org/wiki/Pac-Man) for more information.
[2]Described on page 102 of AIMA, or see http://www.aiai.ed.ac.uk/ gwickler/eightpuzzle-inf.html.