CS 188
Summer 2022

Exam Prep 1B Solutions

Q1. Search

For this problem, assume that all of our search algorithms use tree search, unless specified otherwise.

(a) For each algorithm below, indicate whether the path returned after the modification to the search tree is guaranteed to be identical to the unmodified algorithm. Assume all edge weights are non-negative before modifications.

(i) Adding additional cost $c > 0$ to every edge weight.

	Yes	No
BFS	●	○
DFS	●	○
UCS	○	●

(ii) Multiplying a constant $w > 0$ to every edge weight.

	Yes	No
BFS	●	○
DFS	●	○
UCS	●	○

(b) **For part (b)**, two search algorithms are defined to be **equivalent** if and only if they expand the same states in the same order and return the same path. **Assume all graphs are directed and acyclic.**

(i) Assume we have access to costs c_{ij} that make running UCS algorithm with these costs c_{ij} equivalent to running BFS. How can we construct new costs c'_{ij} such that running UCS with these costs is equivalent to running DFS?

○ $c'_{ij} = 0$ ○ $c'_{ij} = 1$ ○ $c'_{ij} = c_{ij}$

● $c'_{ij} = -c_{ij}$ ○ $c'_{ij} = c_{ij} + \alpha$ ○ Not possible

Breadth-First Search expands the node at the shallowest depth first. Assigning a constant positive weight to all edges allows to weigh the nodes by their depth in the search tree. Depth-First Search expands the nodes which were most recently added to the fringe first. Assigning a constant negative weight to all edges essentially allows to reduce the value of the most recently nodes by that constant, making them the nodes with the minimum value in the fringe when using uniform cost search. Hence, we can construct new costs c'_{ij} by flipping the sign of the original costs c_{ij}.

Q2. State Representations and State Spaces

For each part, state the size of a minimal state space for the problem. Give your answer as an expression that references problem variables. Below each term, state what information it encodes. For example, you could write $2 \times MN$ and write "whether a power pellet is in effect" under the 2 and "Pacman's position" under the MN. State spaces which are complete but not minimal will receive partial credit.

Each part is independent. A maze has **height M and width N**. A Pacman can move *NORTH, SOUTH, EAST, or WEST*. There is initially a **pellet in every position** of the maze. The **goal is to eat all of the pellets.**

(a) Personal Space

In this part, there are P Pacmen, numbered $1, \ldots, P$. Their turns cycle so Pacman 1 moves, then Pacman 2 moves, and so on. Pacman 1 moves again after Pacman P. Any time two Pacmen enter adjacent positions, the one with the lower number dies and is removed from the maze.

$$(MN)^P \times 2^{MN} \times P$$

$(MN)^P$: for each Pacman (the dead Pacmen are "eaten" by the alive Pacman, we encapsulate this in the transition function so that Pacmen in the same position must move together and can only move during the turn of the higher numbered Pacman)

2^{MN}: for each position, whether the pellet there has been eaten

P: the Pacman whose turn it is

$(MN+1)^P \times 2^{MN} \times P$ is also accepted for most of the points. Where $(MN+1)$ also includes DEAD as a position

(b) Road Not Taken

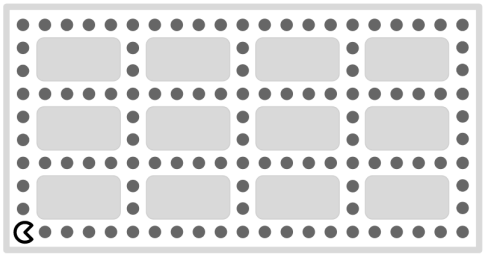
In this part, there is one Pacman. Whenever Pacman enters a position which he has visited previously, the maze is reset – each position gets refilled with food and the "visited status" of each position is reset as well.

$$MN \times 2^{MN}$$

MN: Pacman's position

2^{MN}: for each position, whether the pellet there has been eaten (and equivalently, whether it has been visited)

(c) Hallways



In this part, there is one Pacman. The walls are arranged such that they create a grid of H hallways **total**, which connect at I intersections. (In the example above, $H = 9$ and $I = 20$). In a single action, Pacman can move from one intersection into an adjacent intersection, eating all the dots along the way. Your answer should only depend on I and H.

(note: H = number of vertical hallways + number of horizontal hallways)

$$I \times 2^{2I-H}$$

I: Pacman's position in any one of the intersections

2^{2I-H}: for each path between intersections, whether the pellets there have been eaten. The exponent was calculated via the following logic:

Approach 1:
Let v be the number of vertical hallways and h be the number of horizontal hallways. Notice that $H = v + h$ and $I = v * h$

Each vertical hallway has $h - 1$ segments, and each horizontal hallway has $v - 1$ segments.
Together, these sum to a total of $v(h-1)+h(v-1) = 2vh - v - h = 2I - H$ segments, each of which is covered by a single action.

Approach 2:
Let $H = v + h$ where v is the number of vertical hallways and h is the number of horizontal hallways

$4I$ = Every intersection has 4 paths adjacent to it
$-2v$ = The top and bottom intersection of each vertical hallway has one less vertical path
$-2h$ = The right-most and left-most intersection of each horizontal hallway has one less horizontal path
$4I - 2v - 2h$ must be divided by 2 since you count every path twice in each direction from the intersections which gives you

$$\frac{4I - 2v - 2h}{2} = 2I - v - h = 2I - H$$