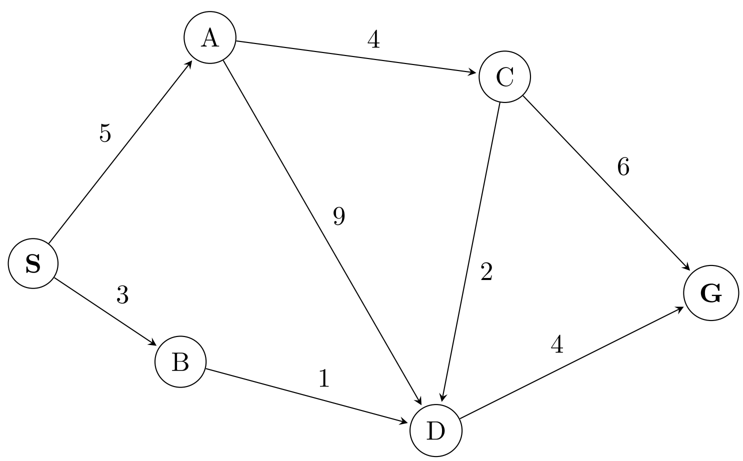
1 Designing & Understanding Heuristics

Today, we will be taking a closer look at how the performance of A^* is affected by the heuristics it uses. To do this, we'll be using the graph below. You may have noticed that no heuristic values have been provided (*Recall:* What is A^* without heuristic values?). This is because we'll be working in pairs to come up with heuristics ourselves!

Please find someone next to you to work with, and decide between yourselves who will design an admissible heuristic and who will design a consistent heuristic. Then, *independently* create these heuristics for the given graph by annotating each node with a heuristic value.

When you have completed your heuristic, exchange your paper with your partner, and work together to answer the questions below.



(a) Write down the path found by the heuristic on the graph above.

This will depend on the provided heuristic. Recall that A^* expands the node in its frontier set that has the lowest $f(n) = g(n) + h(n)$ value.

(b) Work with your partner to come up with a heuristic that's admissible but not consistent.

Heuristics will vary from team to team, and there may be many correct solutions.

(c) Prove that if a heuristic is consistent, it must be admissible.

We can prove that consistency implies admissibility through induction.

Recall that consistency is defined such that $h(n) \leq c(n, n+1) + h(n+1)$.

Base Case: We begin by considering the $n-1$th node in any path where n denotes the goal state.

$$h(n-1) \leq c(n-1, n) + h(n) \tag{1}$$

Because n is the goal state, by definition, $h(n) = h^*(n)$. Therefore, we can rewrite the above as

$$h(n-1) \leq c(n-1, n) + h^*(n)$$

and given that $c(n-1,n) + h^*(n) = h^*(n-1)$, we can see:

$$h(n-1) \leq h^*(n-1)$$

which is the definition of admissibility!

Inductive Step: To see if this is always the case, we consider the $n-2$nd node in any of the paths we considered above (e.g. where there is precisely one node between it and the goal state). The cost to get from this node to the goal state can be written as

$$h(n-2) \leq c(n-2,n-1) + h(n-1)$$

From our base case above, we know that

$$h(n-2) \leq c(n-2,n-1) + h(n-1) \leq c(n-2,n-1) + h^*(n-1)$$

$$h(n-2) \leq c(n-2,n-1) + h^*(n-1)$$

And again, we know that $c(n-2,n-1) + h^*(n-1) = h^*(n-2)$, so we can see:

$$h(n-2) \leq h^*(n-2)$$

By the inductive hypothesis, this holds for all nodes, proving that consistency does imply admissibility!

2 True/False Section

(a) Depth-first search always expands at least as many nodes as A search with an admissible heuristic.

False: a lucky DFS might expand exactly d nodes to reach the goal. A largely dominates any graph-search algorithm that is guaranteed to find optimal solutions.

(b) Assume that a rook can move on a chessboard any number of squares in a straight line, vertically or horizontally, but cannot jump over other pieces. Manhattan distance is an admissible heuristic for the problem of moving the rook from square A to square B in the smallest number of moves.

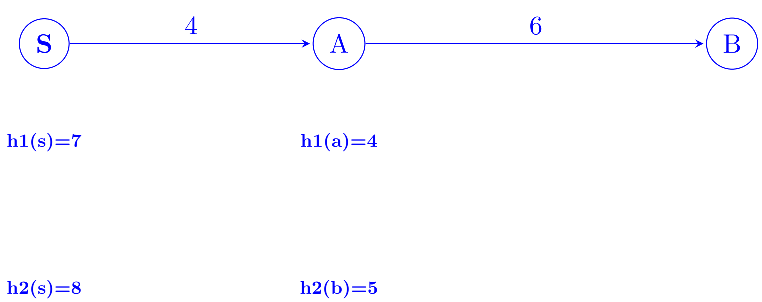
False: a rook can move across the board in move one, although the Manhattan distance from start to finish is 8.

(c) The euclidean distance is an admissible heuristic for Pacman path-planning problems.

True: Euclidean distance will be the minimum cost to travel the path. Thus, it will always be lesser than or equal to the actual cost, making it admissible ($0 \leq h(n) \leq h^*(n)$).

(d) The sum of several admissible heuristics is still an admissible heuristic.

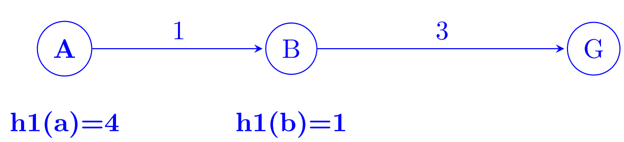
False:



Both of these heuristics (h1 and h2) are admissible, but if we sum them, we find that $h3(s) = 15$ and $h3(a) = 9$. However, this is not admissible.

(e) Admissibility of a heuristic for A^* search implies consistency as well.

False:



$h1(a) > c(a, b) + h1(b)$, which violates the triangle inequality.