CS322 Fall 1999
Module 4 (Search)
Assignment 4

Solution

The aim of this assignment is to learn about the basic search procedures.

Question 1

Consider the graph shown in Figure 1. Suppose the neighbours are given by the following relations:

```
neighbours(s,[a,c,k]).
neighbours(a,[b]).
neighbours(b,[h,g]).
neighbours(c,[d]).
neighbours(d,[e]).
neighbours(e,[f]).
neighbours(f,[g]).
neighbours(g,[]).
neighbours(h,[i]).
neighbours(i,[j]).
neighbours(j,[]).
neighbours(k,[l]).
neighbours(l,[]).
```

Suppose the heuristic estimate of the distance to *g* is:

```
h(a,2).     h(b,3).
h(c,4).     h(d,3).
h(e,2).     h(f,1).
h(g,0).     h(h,4).
h(i,5).     h(j,6).
h(k,5).     h(l,6).
h(s,4).
```

For each of the following search strategies to find a path from *s* to *g*:

(a) Depth-first search

(b) Breadth-first search

(c) Least-cost first search

(d) Best-first search

(e) A* search

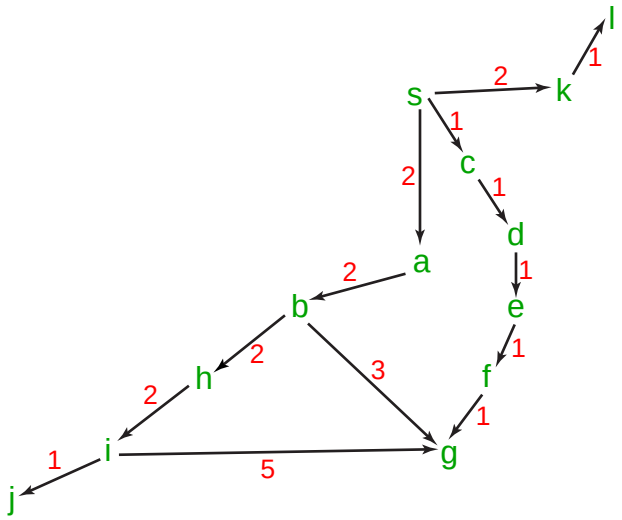


Figure 1: Search Graph

Specify:

i) What is the final path found?

ii) How many nodes were expanded?

iii) Explain why it selected nodes during the search that were not on the shortest path from s to g.

iv) Explain why it may have been led astray in the final solution. (Either state that it found the shortest part, or explain why the shortest path was not found).

Note there are 20 parts to this question. By brief and concise. You can use the search applet available on the web page and at `~cs322/tools/search/search`.

Solution

(a) Depth-first search

i) What is the final path found?
$s \rightarrow a \rightarrow b \rightarrow h \rightarrow i \rightarrow g$.

ii) How many nodes were expanded?
7.

iii) Explain why it selected nodes during the search that were not on the shortest path from s to g.
It selects nodes in order irrespective of where the goal is.

iv) Explain why it may have been led astray in the final solution.
It reports whatever path it finds first; this could be any path depending on the order of the neighbours.

(b) Breadth-first search

i) What is the final path found?
$s \rightarrow a \rightarrow b \rightarrow g$.

ii) How many nodes were expanded?
9.

iii) Explain why it selected nodes during the search that were not on the shortest path from s to g.
It selects all nodes that are two steps from the start node, irrespective of where the goal is, before it expands nodes that are three steps away and finds the goal.

iv) Explain why it may have been led astray in the final solution.
It finds the path with the fewest arcs, not the shortest path.

(c) Least-cost first search

i) What is the final path found?
$s \rightarrow c \rightarrow d \rightarrow e \rightarrow f \rightarrow g$.

ii) How many nodes were expanded?
10

iii) Explain why it selected nodes during the search that were not on the shortest path from s to g.
It explores the paths in order of length, irrespective of where the goal is.

iv) Explain why it may have been led astray in the final solution.
It wasn't; least cost first always finds the shortest path to the goal.

(d) Best-first search

i) What is the final path found?
$s \rightarrow a \rightarrow b \rightarrow g$.

ii) How many nodes were expanded?
4

iii) Explain why it selected nodes during the search that were not on the shortest path from s to g.
It chooses the node closest to the goal, and doesn't take into account the path length from the start node.

iv) Explain why it may have been led astray in the final solution.
Node a was closer to the goal than node c, once it had nodes on the frontier that were close to goal, it never considered c.

(e) A* search

i) What is the final path found? $s \rightarrow c \rightarrow d \rightarrow e \rightarrow f \rightarrow g$.

ii) How many nodes were expanded?
7.

iii) Explain why it selected nodes during the search that were not on the shortest path from s to g.
Node a looked as thought it was on a direct route to the goal. But, the deviation to go to b was greater than the cost of going via c.

iv) Explain why it may have been led astray in the final solution. .
It wasn't; A* always finds the shortest path to the goal.

Question 2

For each of the following, give a graph that is a tree (there is at most one arc into any node), contains at most 15 nodes, and has at most two arcs out of any node.

(a) Give a graph where depth-first search is much more efficient (expands fewer nodes) than breadth-first search.

(b) Give a graph where breadth-first search is much better than depth-first search.

(c) Give a graph where A* search is more efficient than either depth-first search or breadth-first search.

(d) Give a graph where depth-first search and breadth-first search are both more efficient than A* search.

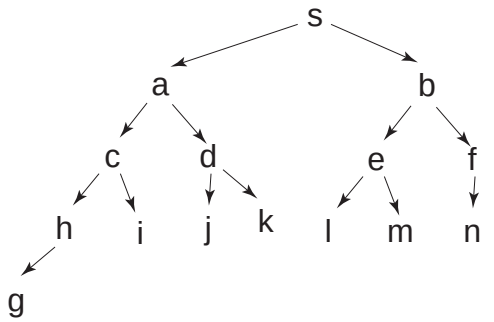
You must draw the graph and show the order of the neighbours (this is needed for the depth-first search). Either give the arc costs and heuristic function or state explicitly that you are drawing the graph to scale and are using Euclidean distance for the arc costs and the heuristic function.

Solution

In all of these graphs, we assume that we are on a plane, with Euclidean distance (straight-line distance) as the arc cost and as the heuristic function. We also assume that the neighbours are ordered from left to right. The start node is *s* and the goal node is *g*.

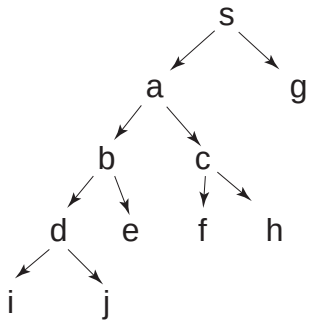
(a) Give a graph where depth-first search is much more efficient (expands fewer nodes) than breadth-first search.

Here depth-first search expands every node, whereas breadth-first search expands three nodes:



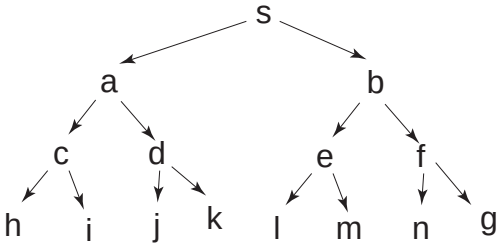
(b) Give a graph where breadth-first search is much better than depth-first search.

Here depth-first search expands every node, whereas breadth-first search expands three nodes:



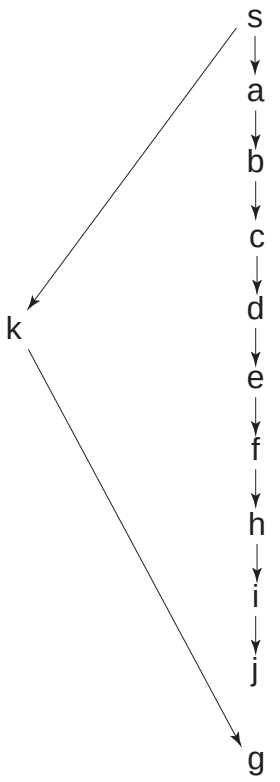
(c) Give a graph where A* search is more efficient than either depth-first search or breadth-first search.

Here depth-first search and breadth-first search expand every node, whereas A* search expands 4 nodes.



(d) Give a graph where depth-first search and breadth-first search are both more efficient than A* search.

Here depth-first search expands three nodes, breadth-first search expands 4, yet A* search expands every nodes.



Question 3

For each question in this assignment, say how long you spent on it. Was this reasonable? What did you learn?