Foundations of Artificial Intelligence

Prof. Dr. B. Nebel, Prof. Dr. W. Burgard
B. Frank, A. Karwath, G. Röger
Summer Term 2009

University of Freiburg
Department of Computer Science

Exercise Sheet 2

Due: Tuesday, May 12, 2009

Exercise 2.1 (Search algorithms)

Prove each of the following statements:

(a) Breadth-first search is a special case of uniform-cost search.

(b) Breadth-first search, depth-first search, and uniform-cost search are special cases of best-first search.

(c) Uniform-cost search is a special case of A* search.

Exercise 2.2 (Path costs)

So far, we have assumed in the lecture that path costs are not negative. In this exercise, we would like to discuss negative path costs and cyclic paths.

(a) Suppose that a negative lower bound $c < 0$ is placed on the cost of any given step, i.e. negative costs are allowed, but the cost of a step cannot be less than c. Does this allow *uniform-cost* search to avoid searching the whole tree? If so, why?

(b) Suppose that there is a set of operators that form a loop, so that executing the set in some order results in no net change to the state. If all of these operators have negative costs, what does this imply about the *optimal* behaviour for an agent in such an environment.

Exercise 2.3 (A* search)

Trace the operation of A* search in the following 8-puzzle configuration:

2	8	3
1	6	4
7		5

Goal State:

1	2	3
8		4
7	6	5

Show the sequence of search nodes the algorithm will consider and the f, g, and h score for each node when used with the Manhattan distance heuristics and with the "Misplaced Tiles" heuristics. How does the heuristic influence the search?

Exercise 2.4 (Heuristic functions)

Suppose we introduce a weight α to the A* heuristic function, so that $f(n) = g(n) + \alpha h(n)$. How does the A* algorithm behave in general, if:

- $\alpha = 0$
- $\alpha > 1$?

How does $\alpha > 1$ influence the complexity and optimality of the search? Does a positive weight improve the performance of A*, i.e. does it make sense? Why?

The exercise sheets may and should be worked on in groups of three (3) students. Please fill the cover sheet[1] and attach it to your solution.

[1] `http://www.informatik.uni-freiburg.de/~ki/teaching/ss09/gki/coverSheet-english.pdf`