Einführung in die Künstliche Intelligenz 2013S, 2.0 VU, 184.735

Exercise Sheet 1 - Search

You have to tick the prepared exercises in TUWEL at the latest before Friday, 19th April 2013, 13:00 (Exercise Sheet 1). Belated submission will be ignored.

Exercise 1 (Uninformed Search):

a) Let $b > 1$ be the maximal branching degree in the search tree and let d be its depth. Estimate the number of nodes, $n_{bfs}(d)$, generated during a breadth-first search with depth d. Show that $n_{bfs}(d) = O(b^d)$ and estimate the constant c_{bfs}. **(2 pts)**

b) Let $b > 1$ be the maximal branching degree in the search tree and let d be its depth. Estimate the number of nodes, $n_{dfid}(d)$, generated during a depth-first-iterated-deepening search with depth d. Show that $n_{dfid}(d)$ is $O(b^d)$ and estimate the constant c_{dfid}. What can you say about the overhead induced by dfid? **(2 pts)**

c) Describe a state space in which dfid is much worse than dfs, e.g., $O(n^2)$ vs. $O(n)$. **(1 pt)**

Exercise 2 (Informed Search - heuristics):

a) Prove that, if any given heuristics h_1, h_2 are both monotone (consistent), then

$$h(n) = \max(h_1(n), h_2(n)).$$

is also monotone (consistent). **(2 pts)**

b) Let h_1, h_2 both be admissible heuristics. Check whether the following heuristics $h(h_1, h_2)$, which are combinations of h_1, h_2, are also admissible. If $h(h_1, h_2)$ is not admissible, then estimate intervals for which admissibility is given.

1. $h(h_1, h_2) = \frac{h_1+h_2}{c^2-h_1 \cdot h_2}$ $\qquad c > h_1, c > h_2$ **(2 pts)**
2. $h(h_1, h_2) = h_1 \cdot h_2$ **(1 pt)**

Exercise 3 (A^* -search and miscellaneous):

a) Let $f(n) = c_g g(n) + c_h h(n)$ be an evaluation function, where c_g, c_h be constants.

1. Define $c_g, c_h, h(\cdot), g(\cdot)$ such that A^* with this evaluation function is bfs. **(1pt)**
2. Define $c_g, c_h, h(\cdot), g(\cdot)$ such that A^* with this evaluation function is dfs. **(1pt)**

b) Which of the following statements are true and which are false? Explain your answers.

1. Depth-first search always expands at least as many nodes as A^* search with an admissible heuristic. **(1 pt)**
2. 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 smalles number of moves. **(1 pt)**
3. Breadth-first search is a special case of uniform-cost search. **(1 pt)**