CMPT 310, Spring 2019, Written Assignment 02

Due date: January 21, 2019

Problem 1. Which of the following are true and which are false? Explain your answers.

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

(b) $h(n) = 0$ is an admissible heuristic for the 8-puzzle.

(c) A^* is of no use in robotics because percepts, states, and actions are continuous.

(d) Breadth-first search is complete even if zero step costs are allowed.

(e) 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 rock from square A to square B in the smallest number of moves.

Problem 2. The iterative lengthening search is an iterative analog of uniform cost search. The idea is to use increasing limits on path cost. If a node is generated whose path cost exceeds the current limit, it is immediately discarded. For each new iteration, the limit is set to the lowest path cost of any node discarded in the previous iteration.

(a) Show that this algorithm is optimal for general path cost.

(b) Consider a uniform tree with branching factor b, solution depth d, and unit step costs. How many iterations will iterative lengthening require?

(c) Now consider step costs drawn from the continuous range $[\epsilon, 1]$, where $0 < \epsilon < 1$. How many iterations are required in the worst case?

Problem 3 (Optional). Describe a state space in which iterative deepening search performs much worse than depth-first search (for example, $O(n^2)$ vs. $O(n)$).

Problem 4 (Optional). Prove each of the following statements, or give a counterexample:

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

(b) Depth-first search is a special case of best-first tree search.

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

Problem 5 (Optional). The heuristic path algorithm (Pohl, 1977) is a best-first search in which the evaluation function is $f(n) = (2 - w)g(n) + wh(n)$. For what values of w is this complete? For what values is it optimal, assuming that h is admissible? What kind of search does this perform for $w = 0$, $w = 1$, and $w = 2$?