Artificial Intelligence I 2004 – Paper 5 Question 6

For 2004 only, the Question for the Part II Artificial Intelligence course was the same as that for the Part IB course.

Context: the question is from the section of the course dealing with problem-solving via heuristic search. It involves the description of the general properties of a problem making this form of AI an appropriate technique to be applied. It then requires a description of the $A^\star$ algorithm and proofs of its central properties.

1. The problem needs to be capable of being represented as a set of *states*, with *actions* causing transitions between states. It should have a well-defined *start state*, and one or more *goal states*, and there must be a well-defined way in which to test whether a given state is a goal. Potential solutions are in the form of *paths*, which should begin with the start state and end with a goal state and consist of sequences of states where each can be reached from the preceding state in the path via an available action. Typically, a function p called the *path cost* should be available denoting the desirability of a given path. Heuristic algorithms will typically make use of a *heuristic function* h which estimates the cheapest cost distance from a given state to a goal.

2. An algorithm is complete if it is guaranteed to find a solution. That is, a path from the start state to a goal state. An algorithm is optimal if it is guaranteed to find the *best* solution in terms of path cost.

3. A heuristic is admissible if it never overestimates the cost to the goal. It is monotonic if when added to the path cost giving the combined cost $f(n) = p(n) + h(n)$ then $f(n)$ never decreases along a path.

4. Beginning with the start state, expand it to generate all possible successors. In general search algorithms keep a queue of un-expanded states ordered according to some criterion, and successively expand states in that order. At each expansion a test is made to see whether a goal state has been obtained. If so, the search terminates, otherwise we continue. The $A^\star$ algorithm works in precisely this way, using the cost

$$f(n) = p(n) + h(n)$$

to prioritise the expansion of states, where states with low $f(n)$ are expanded before states with high $f(n)$.

5. Let G be the optimal goal state, where

$$f(G) = f' = p(G)$$

as obviously $h(G) = 0$ for a goal state. Let H be a suboptimal goal such that

$$f(H) = p(H) > f'$$

If the algorithm were ever to select H for expansion then the search would stop, so we need to show this can not happen. Assume there is a state n which is at present a leaf in the search tree on an optimal path to G. As h is admissible

$$f' \geq f(n) \tag{1}$$

and if H is chosen instead of n then

$$f(n) \geq f(H) \tag{2}$$

Combining (1) and (2) gives

$$f' \geq f(H) = p(H)$$

which is a contradiction as we know that H is suboptimal. In order for the procedure to be complete, we note that as states are expanded according to increasing $f(n)$ we must obtain a goal as long as we can not make a path which has infinitely many states with $f(n) < f'$. So we require that: (1) no state has an infinite branching factor, and; (2) there is no path that is an infinite sequence of states with a finite path cost.