

William Daniels

CSCI 4630

Linguistic Geometry

Homework #9 04/04/15

1. First, the MV and CUT for simple hill climbing: For MV, the algorithm would check the heuristic value of each triple (P, X, Y) produced by MOVE, and then whichever heuristic value was closest (but still better) than the current evaluated value would be returned. -1 would be returned if nothing was greater than the current value. CUT for simple hill climbing would be a simple check to see if $MV = -1$, since that would tell whether or not it had 'plateaued' (reach either a local or global max). This would be susceptible to not finding the global maximum, but that's simply a trait of the simple hill climbing. An example of this search, we'll use the search tree for the robotic vehicles with alternating serial motions. (see page 11 of lecture 19 for the tree). For the first move (h8-g7) the function MOVE would take the heuristic value and determine that it should move to g7 first, as opposed to c7, since the heuristic value will be higher for the h8-g7 move. Now, the next step, would be much the same, both moves would return the same heuristic value, however, and the function would choose one at random. This is the only 'plateau' in the tree, so the rest of the program would fall out normally, and simple hill climbing would produce some optimal results.
2. Second the MV and CUT for steepest-ascent hill climbing it's very similar to the one for simple hill climbing, except the goal state would have a defined heuristic 'value' of some sort (aka: the solution is known), and then at each step MV would check all of the current successive triples produced by MOVE, and then would pick the one which was closest to the solution value. (aka, the highest-valued 'steepest' choice). The function CUT would be identical for this as before, since the function MV would return -1 if no closer node to the solution can be found. The example is much the same, except it would choose the top tree, since the solution is 'closer'.
3. For the grammar of trajectories, the MV would be produced by determining a value in 3 ways.
 - 1) Positional advantage, how close to a given 'target' (physical number of steps)
 - 2) The material advantage, which is how many pieces have been won/lost/will be lost in the next move or two. and
 - 3) safety, if the piece can move to a given location without being captured, or causing another piece to be captured. Now, this would produce a numerical value for the trajectory 'move' at each step. CUT will comprise of a few things: 1) if the numerical value of the trajectory move is below a certain threshold, it will be CUT. If the timer value for the trajectory to reach the negation it is reduced below 0, then it will be CUT (Frozen). Following the rules of how to 'freeze' zones with the dynamic zone, makes a lot of these automatically fall off.