

Artificial Intelligence

Mark Anderson

Homework 3

February 13, 2019

On page 68, we said that we would not consider problems with negative path costs. In this exercise, we explore this decision in more depth.

1. Suppose that actions can have arbitrarily large negative costs; explain why this possibility would force any optimal algorithm to explore the entire state space.
 - If there exists some path in the state space that is not bound by a negative path cost limit, the optimal algorithm would be required to exhaust the entirety of the state space. This is because the algorithm will never know if there is a large enough negative path cost, to offset the path cost required in getting there.
2. Does it help if we insist that step costs must be greater than or equal to some negative constant c ? Consider both trees and graphs
 - Tree: This changes the optimal search for a tree because at any point, you know the upperbound of the largest negative path cost, and the algorithm can determine whether or not it is optimal to offset the cost of searching some negative path that is equal to the constant C .
 - Graph: The graph is a special case because graphs are able to contain cycles, whereas tree cannot. If there exists any cycle with negative paths in the graph, an optimal algorithm will search that cycle forever in order to get the lowest path cost.
3. Suppose that a set of actions forms a loop in the state space such that executing the set in some order results in no net change to the state. If all of these actions have negative cost, what does this imply about the optimal behavior for an agent in such an environment?
 - The same answer as above, if a loop/cycle exists in the state space that does not affect the state, the algorithm will always traverse that loop forever in order to reach the lowest path cost.