Kote Mushegian and Clara Hunnewell

Artificial Intelligence

Stephen Majercik

12/7/2016

MDP Lab

*1. Find and describe three settings of the command-line arguments that produce policies that result in the following paths (i.e. one setting for each of these three): Also, explain, in English, why you think these parameters are producing the particular behavior.*

1. *a direct path that goes between the negative reward squares in one or both directions*

0.99 1e-6 0.5 1 -1 -0.04

In this setting, we are keeping all of the arguments the same except for the probability of key loss. We are setting the probability of key loss equal to 1 which means that if the agent goes through the Lose-Key? square, the agent will lose the key with 100% probability.  Thus, the agent will want to avoid the Lose-Key square at all costs because it needs the key to get the positive reward and the higher expected utility. The agent takes the direct path also because of the negative step cost or penalty for more indirect paths. As a result, in order to get the key and to avoid the accumulation of the negative step costs, it will risk the direct path between the negative reward squares, and, on the way back, it will risk the direct path between the negative reward square so as to avoid the possibility of landing in the Lose-Key? square, which would decrease its utility.

1. *an indirect path that goes through the Lose-Key? square in one or both directions,*

0.99 1e-6 0 1 -1 0

In this settings, we have made two changes.  We want the agent to pass through the Lose-Key? square. However, passing through the Lose-Key? square with the original settings has negative repercussions since the agent loses the key with a 50% probability, and the agent needs the key to get the positive reward that will increase its utility.  As a result, we have set the probability of the agent losing the key when it passes through Lose-Key? equal to zero so that, even if the agent passes through that square, it will not lose the key and its utility will not be diminished. The other change we made was to set the step cost equal to zero. In the original setting, the agent avoided/was punished for indirect paths because the longer the path the agent took, the greater the accumulation of negative step costs.  We did not want our agent being discouraged from taking a long indirect path and thus made it so that, in terms of step cost, the agent is indifferent between the indirect and the short path. The negative terminal rewards further discourage the agent from taking the direct path and result in the agent taking the indirect path through Lose-Key?.

1. *an in-between path that goes through the square between the Lose-Key? square and the leftmost -1 square in one or both directions. Note that while you are allowed to change the magnitudes of the rewards and penalties in the reward and penalty squares, they should still have the same sign as they did in the original version of the problem.*

0.99 1e-6 1 1 -1 0

In this setting, the probability of losing a key is changed to 1 which means that the agent will always lose its key if it passes through Lose-Key?, and the step-cost is set to zero.  This will result in the agent passing between the Lose-Key? square and the leftmost negative reward on the way back (once it already has the key). Since going through the Lose-Key square results in the key being lost and a lower utility, the user will avoid it.  The agent will choose the in-between path instead of the direct path because there is no penalty for more indirect paths and because it will be less likely to land in a -1 reward square since it is only passing next to one of them (probability of .1 of negative terminal reward) as opposed to two (probability of .2 of getting negative terminal reward).

*2. Find a case in which changing the discount factor, while keeping everything else unchanged, produces a drastic change in the overall objective of the policy. Describe the change you made and the resulting change in the policy objective. Why did your change cause the change in the policy objective.*

Changing the discount factor to .01 produces a drastic change in overall objective of the policy.  The policy in almost all of the states in the top half of the graph is to go North whereas the policy in the bottom row of the graph is to go east always. This will result in the agent going east toward the positive terminal reward state even before getting the key which unlocks its terminal reward.

Making the discount factor 0.01 means that expected utilities which are “further away” (realized in a greater amount of moves) will not have as much value to the agent as the expected utilities of states that are closer (realized in few moves) because those that are further away are being greatly discounted.  This results in a faster convergence (and indeed, with a discount factor of 0.01, convergence occurs after 3 iterations compared to 50 iterations when there was a discount factor of 0.99) because a heavier weight is given to the expected values that can be attained in few moves and increasing iterations does little to change these expected values; the values from later iterations are being discounted so greatly that they have little pull in determining the best action at the current square and thus the iterations are terminated.  As a result, the agent may miss some of the long-term benefits/costs of certain moves because they are being so greatly discounted as to become insignificant in the calculation of the expected utility of that move.

The 0.01 discount factor caused the change in policy objective to go east before attaining the key because, since expected utilities that are “far away” (require many moves to attain) are being discounted so greatly, the agent operates with a very short-term horizon.  It does not recognize the benefit of first going to get the key and then going to the positive reward terminal state (which would take a lot of moves) and instead is drawn towards the 0 step cost at the positive terminal state which can be reached in fewer moves.

(textbook pg. 654)