Question 1: How many iterations does it take for the Value Iteration algorithm to converge? In an output text file list the optimal values ($V*$ for each state).

Answer 1:

Number of iterations to converge: 5

The optimal Values are:

V*(s1) = 51.2

V*(s2) = 64

V*(s3) = 0

V*(s4) = 64

V*(s5) = 80

V*(s6) = 100

Question 2: Assume we start in state s1, give the states that form the optimal policy ($\pi*$) to reach the terminal state (s3).

Answer 2:

Optimal Policy grid:

s2 | s5 | end|

s5 | s6 | s3 |

This grid shows the states that form the optimal policy when starting at any index.

The state represented in each block shows the action needed to get to form an optimal policy.

Thus for a state to start at s1 and end at s3 the actions it would take (states it would go through) would be:

s1->s2->s5->s6->s3

Question 3:Is it possible to change the reward function function so that $V*$ changes, but the optimal policy ($\pi*$) remains unchanged?

Answer 3:

Yes, If you double the the immediate rewards then it causes the V* values to double aswell. This will result in the same optimal policy being produced.

The below V* values is produced when the immediate rewards are doubled:

V*(s1) = 102.4

V*(s2) = 128

V*(s3) = 0

V*(s4) = 128

V*(s5) = 160

V*(s6) = 200

which results in the same optimal policy grid as if they werent doubled:

s2 | s5 | end|

s5 | s6 | s3 |

(This was tested by running the code with doubling the immediate rewards).

----------End----------