Theory homework 9

Shaun Toh 1002012

July 22, 2019

1 Task 1

Required to prove that this converges for all values of $Q_0(s, a)$

$$Q_{k+1}(s, a) = R(s, a) + \gamma \sum_{s'} P(s'|s, a) max_{a'} Q_k(s', a')$$

1. in this case, s' and a' are the previous states and actions.
2. Rewards depend only on state action pair. i.e $R(s, a, s') = R(s, a)$ so, previous state does not matter

we first begin by noting that $R(s, a)$ is a constant, and $0 < \gamma < 1$.
$\gamma \sum_{s'} P(s'|s, a) max_{a'} Q_k(s', a')$, if expanded, becomes a geometric progression. We begin by ignoring the $R(s, a)$ term in the current equation since it is a series of constants that will appear based off $P(s'|s, a)$. This leaves us with

$$Q_{k+1}(s, a) = \gamma \sum_{s'} P(s'|s, a) max_{a'} Q_k(s', a')$$

As stated above, $0 < \gamma < 1$. We then note that $\sum_{s'} P(s'|s, a) max_{a'} Q_k(s', a')$ is yet another approximation for the expected value of the next state. Since we seek convergence, infinity will eventually result in all possible iterations of $Q_k(s', a')$ being explored repeatedly. This means that the expected values of each of the future states are also adjusted accordingly, and used to update the value of the current state action pair. Due to the gamma value, it then converges, no matter the value of Q_0, especially because gamma reduces the weightage of previous Q_0 values with each iteration, pushing it closer to its actual values.

2 Task 2

The γ discounted reward is just the sum of the geometric progression sums of both 2 and 3.
Since $2 + 3\gamma + 2\gamma^2 + 3\gamma^3 + 2\gamma^4 + ...$
The sum is hence:

$$3 \times \frac{\gamma}{1 - \gamma^2} + \frac{2}{1 - \gamma^2} = \frac{110}{9}$$

3 Task 3

3.1 DFS

Assuming that lowest alphabetical order means C goes before B,
Steps (Starting from A.):

1. Fringe = [AB,AC]
2. Fringe = [AB,ACE,ACF]
3. Fringe = [AB,ACE,ACFG,ACFH]
4. Fringe = [AB,ACE,ACFG]
5. Fringe = [AB,ACE,ACFGX,ACFGZ]

3.2 BFS

Steps (Starting from A.):

1. Fringe = [AC,AB]
2. Fringe = [ABZ, ABF,AC]
3. Fringe = [ACE,ABZ,ABF]
4. Fringe = [ABFH,ABFG,ACE,ABZ]
5. Fringe = [ABZX,ABZG,ABFH,ABFG,ACE]

3.3 UCS

1. Fringe = [(AC,1),(AB,10)]
2. Fringe = [(ACE,3),(ACF,8),(AB,10)]
3. Fringe = [(ACEH,5),(ACF,8),(AB,10)]
4. Fringe = [(ACEHF,7),(ACF,8),(AB,10),(ACEHX,12)]
5. Fringe = [(ACEHFB,8),(ACF,8),(ACEHFG,9),(AB,10),(ACEHX,12)]
6. Fringe = [(ACF,8),(ACEHFG,9),(ACEHFBX,9),(AB,10),(ACEHFBZ,10),(ACEHX,12)]
7. Fringe = [(ACFB,9),(ACEHFG,9),(ACEHFBX,9),(ACFB,10),(AB,10),(ACEHFBZ,10),(ACEHX,12)]
   Terminate here due to a goal path being same cost as other paths.