TPC5

December 3, 2018

In [22]:
```
import numpy as np


print("1a)")

Qt = 3.08
c = 1.0

#(4, -b r)
Q4r = np.array([
    [3.23 , 3.38, 3.25, 3.22]
])

#(5, -b r)
Q5r = np.array([
    [3.08, 3.25, 3.57, 3.22]
])

Qnew = Qt + 0.1 * (c + 0.99 * np.min(Q5r) - Qt)


print("Q-Learning Q-value = %s" % Qnew)

print("\n1b)")

Qt = 3.08

Qnew = Qt + 0.1 * (c + 0.99*Q5r[0][3] - Qt)

print("Sarsa Q-value = %s" % Qnew)


print("\n1c)\n")

print("Off-policy refers to reinforcement learning methods that learn the value of a p
print("On-policy refers to reinforcement learning methods that learn the value of the
print("SARSA is more stable than Q-learning, as long as learning policy changes smoot
print("The difference between the two update computations: the Q-learning chooses the
```

```
        print("While the SARSA algorithm it is based on the the policy it is following, in thi
```

```
1a)
Q-Learning Q-value = 3.17692

1b)
Sarsa Q-value = 3.19078

1c)

Off-policy refers to reinforcement learning methods that learn the value of a policy while foll
On-policy refers to reinforcement learning methods that learn the value of the policy that the
SARSA is more stable than Q-learning, as long as learning policy changes smoothly with Qt
The difference between the two update computations: the Q-learning chooses the minimum value o
While the SARSA algorithm it is based on the the policy it is following, in this case coded Q5
```