Authors:
i6197660: Rodrigo Emmanuel Brechard Alarcia
i6193659: Khasbulat Kerimov

# Hanoi Tower using Q-Learning

## The assignment

Solve a Tower of Hanoi problem with three pins 1, 2 and 3 and two disks A and B. Disk A is larger than B. The goal is to move the two disks to pin 3 such that the lager disk A is at the bottom and the smaller disk B is at the top. Reaching this goal has a reward of: 100. The agent can move only one disk at the time. We do not forbid to put the larger disk A on top of the smaller disk B, but there is a penalty for doing this; i.e., a reward of: -10. To encourage the agent to solve the problem in a minimal number of steps, all actions that do not result in the goal state or the penalized state have a reward of: -1.

Our agent can make mistakes. When moving a disk from pin *i* to pin *j*, the agent may actually put the disk on pin *k* where $k \neq i$ and $k \neq j$. The probability of making a mistake is: 0.1.

Find the optimal policy for the MDP using **Reinforcement Learning**, assuming that the agent does not know the above specified transition function and reward function. You may ignore the *exploitation* of the results that have been learned.

After reaching the absorbing state, continue the learning process in a randomly selected other state.

## State description

The states have been named with 4 characters, an integer after ‘b’ represents the pin where the **big** disk is placed. An integer after ‘s’ represents the pin where the **small** disk is placed. Therefore, the pattern is always letter + number + letter + number. It is important to notice that when both disks are in the same pin, the order of letters indicate which one is on top (the second is on top of the first one), if they are in different pins then the couple will always indicate where the big disk is. Given this policy, the 12 possible states are named:

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1s1 | 1s2 | 1s3 | 2b2 | 3b3 | 3s3 | 2s3 | 3s3 | 2s2 | 3s1 | 2s1 | 1b1 |

## Actions description

There are six possible actions, moving the small disk to the pin 1, 2 or 3 and the same for the big disk. With the constraint that if both disks are in the same pin only the disk on the top can be moved, therefore, there is a maximum of 4 possible actions at each state.

Authors:
i6197660: Rodrigo Emmanuel Brechard Alarcia
i6193659: Khasbulat Kerimov

The second constraint is that null moves are not considering meaning that if the small disk is in the pin 1, moving it again to the pin 1 is not considered as a move.

To name the actions, the pattern is similar to the states ‘s1’ means moving the small disk to the pin 1 and ‘b2’ means moving the disk to the pin 2. Therefore, the possible actions are:

| b1s1 | b1s2 | b1s3 | s2b2 | s3b3 | b3s3 | b2s3 | b3s3 | b2s2 | b3s1 | b2s1 | s1b1 |
|---|---|---|---|---|---|---|---|---|---|---|---|

## Results

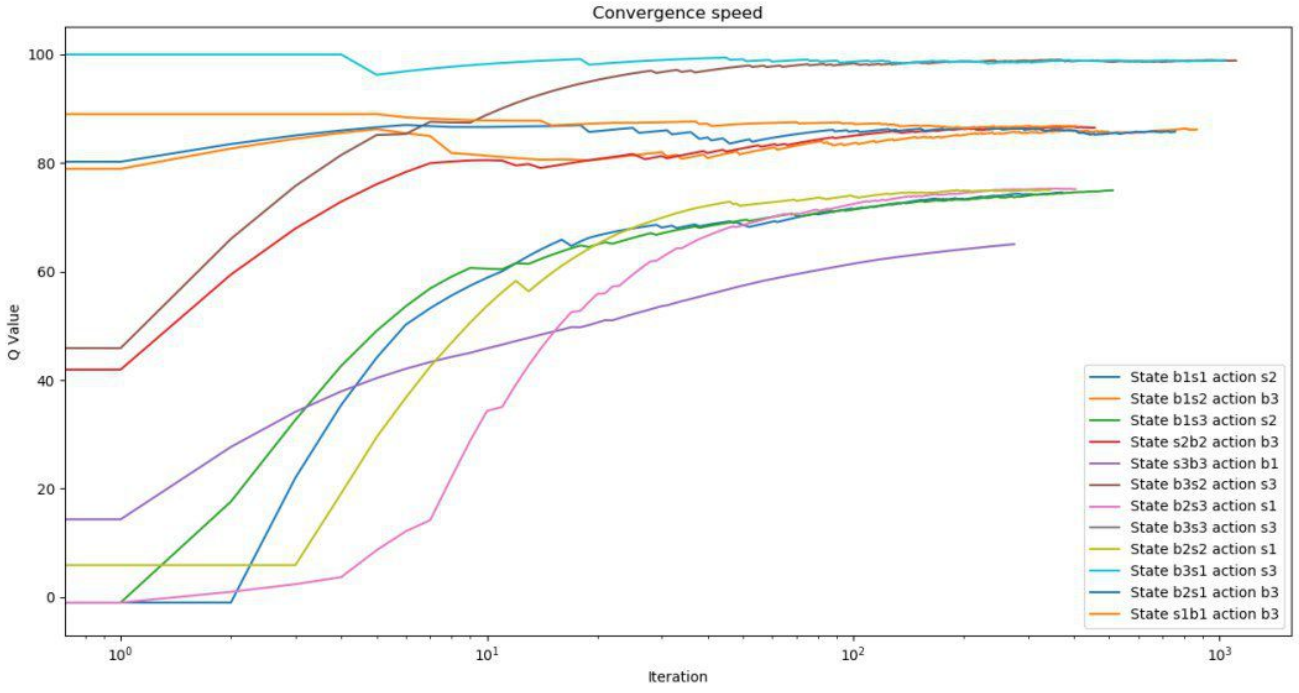
The results for Q-Learning algorithm is presented in the table. The probability of exploring is decreasing linearly from 1 to 0 each step throughout the algorithm. The number of iterations is 10 000. The learning rate function was chosen in a way to meet the convergence requirements. The Q-Learning policy results are the same as for the value and policy iterations.

| State | Policy | Max(Q) |
|---|---|---|
| b1s1 | s2 | 74.97 |
| b1s2 | b3 | 86.08 |
| b1s3 | s2 | 75.04 |
| s2b2 | b3 | 86.83 |
| s3b3 | b1 | 64.02 |
| b3s2 | s3 | 98.73 |
| b2s3 | s1 | 72.89 |
| b3s3 | Absorbing state | |
| b2s2 | s1 | 72.42 |
| b3s1 | s3 | 98.67 |
| b2s1 | b3 | 85.58 |

Authors:
i6197660: Rodrigo Emmanuel Brechard Alarcia
i6193659: Khasbulat Kerimov

| | | |
|---|---|---|
| s1b1 | b3 | 86.72 |

In the following graphs the convergence speed of Q-Values for each pair State - Optimal Policy is shown in logarithmic scale.




## Discussion

In Q-Learning it is used a mix of exploration and exploitation. Since the transition function is not known it is noticeable that at the beginning there should be more exploration than exploitation. Exploration consists on randomly choosing the possible actions in each state and exploitation is executing the action which has the highest Q-Value for that state, using what has been already learnt.

As stated by Michael Kearns and Satinder Singh (Finite-Sample Convergence Rates for Q-Learning and Indirect Algorithms) “the order of $(N\log(1/c)/c^2)(\log(N + \log\log(l/c))$ transitions are sufficient for the algorithm to come within c of the optimal policy”.