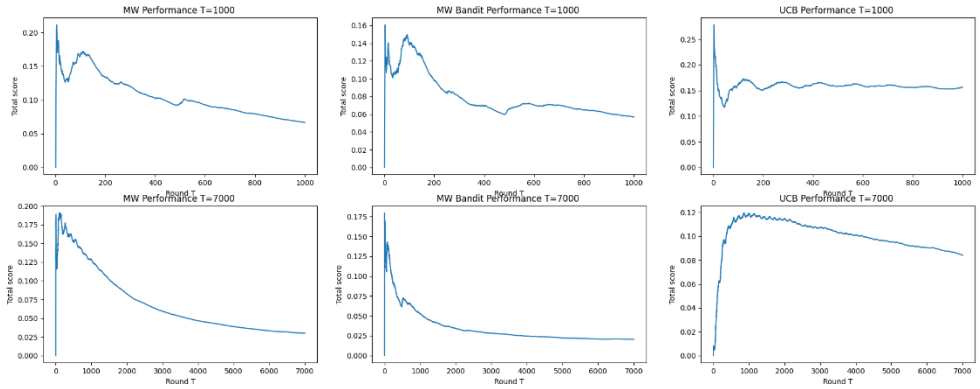
Reinforcement Learning: Homework 2

Due Date: 28 April 2023

Ιωαννίδης Χρήστος 2018030006

Algorithm submitted in .py format.
In order for the algorithm to execute correctly please place the csv file in the same directory.

Below are displayed the outputted regret-time graphs for each algorithm (Please zoom in for more detail):



The regret above is calculated by subtracting the cumulative loss of the best server until in each turn to the cumulative loss of the server chosen by the algorithm (until the same turn).

Observations:

- The MW expert algorithm and the MW bandit algorithm exhibit sublinear regret just as expected.
- The algorithms with the larger horizon T=7000 seem to bear clearly better results than their T=1000 counterparts. This is due to the algorithms given more time to learn
- Even though theoretically the MW bandit algorithm is supposed to be performing worse that the MW expert setup (due to less information being given), in many cases it seems to be performing slightly better. This is probably due to the **η** parameter chosen for the MW expert setup not being optimal for this case (even though the one in the notes is used in the implementation, it can still be optimised further)

- The UCB algorithm is observed to have far worse performance than the other 2 algorithms. This is due to the lack of the probabilistic element required in these kinds of adversarial setups.
- The main adjustment made to the UCB algorithm in order to work in this setup was changing the loss factor into a gain factor. More specifically **gain=1-loss**. It is observed that since all the possible losses are smaller than 1 we just have ‘reverse’ them so that bigger is better in order to be able to add them up.