DEEP REINFORCEMENT LEARNING FOR DYNAMIC SPECTRUM ACCESS

Sanket Deshpande [3050]
Shikhar Tiwari [3047]

Supervisor: Mr. Himanshu Agrawal

ISSUES

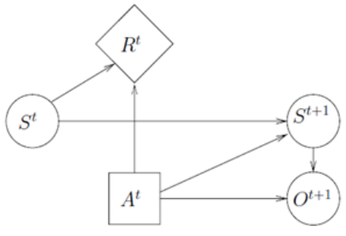
- The increase in wireless devices has led to crowded spectrum.
- Almost the entire useable spectrum is occupied
- Studies demonstration that a big slice of licensed spectrum is unused for any position or time.
- Collisions and errors in spectrum sensing happen in OSA network

GOAL

Develop a distributed learning algorithm for dynamic spectrum access that can effectively use vacant spectrum time slots also called as white-space hunting

ARCHITECTURE

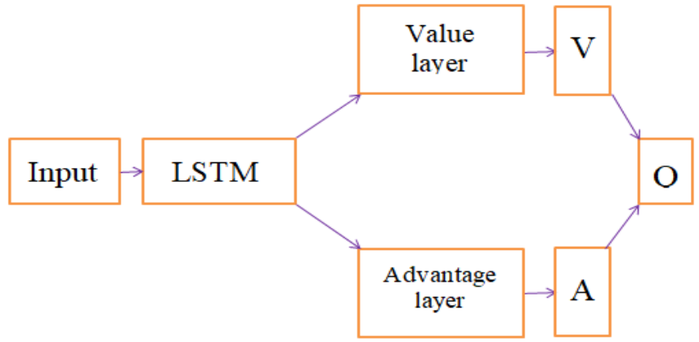
- Network consists of users with a set N = {1, 2... N} and shared independent channels of a set K = {1, 2... K}
- For every time slot in the time period, an agent is permitted to choose only one channel for transmission with some given positive probability.
- After every individual time slot, user receives a binary observation of o_n(t) i.e. 0 or 1, indicating whether the packet was positively carried or not (i.e., ACK indication) in which each user has attempted to transmit a packet. If the packet was delivered positively, we receive o_n (t) = 1. Else, if the broadcast was a failure then o_n (t) = 0.

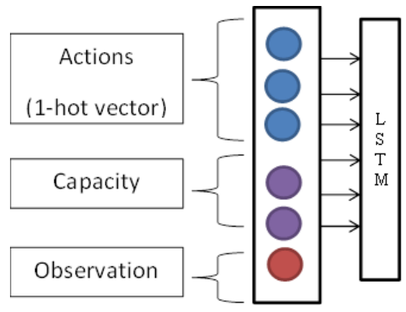


$$R_n = \sum_{t=1}^{T} \gamma^{t-1} r_n(t)$$

DRL

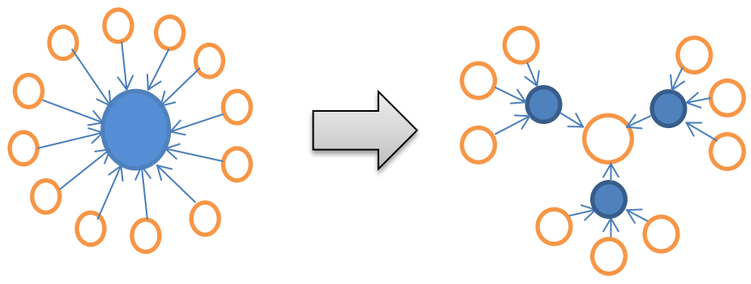
Deep Reinforcement Learning (DRL) Q-learning and Q-learning is a reinforcement learning method that purposes at finding good policies for dynamic programming problems. It has the ability to gauge the predictable utility amongst all the actions that are available without any necessary previous knowledge around the model of the system



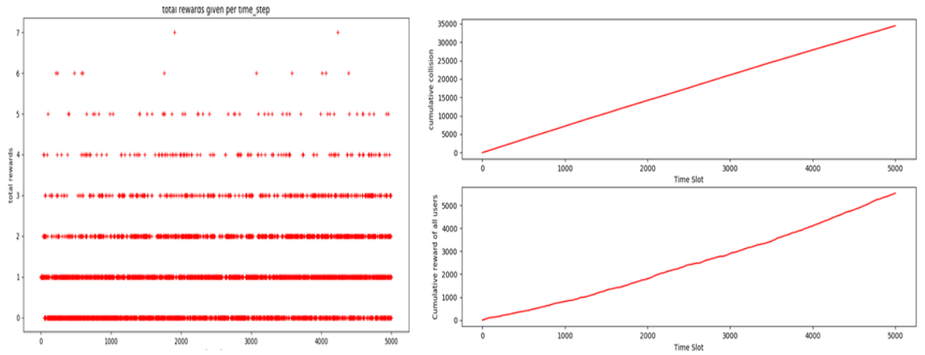


DQN gives Q values which gives probability to take an action

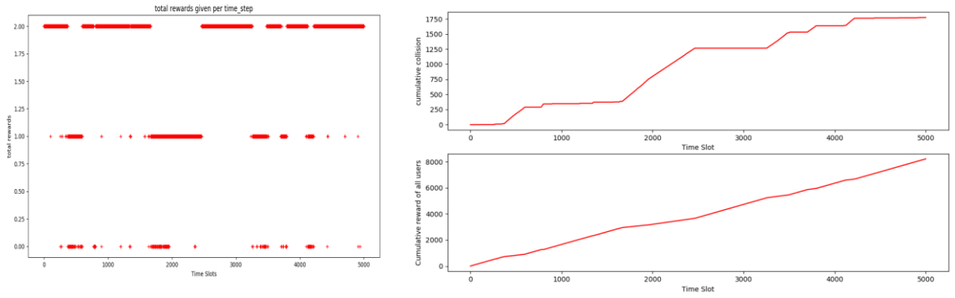
TOPOLOGY



RESULTS



12 users and 8 channels



4 clusters of 3 users and 2 channels

CONCLUSION

We implemented DRL on this problem to get efficient throughput and to reduce the collisions on network. Convergence period of this method increases with number of users and it becomes impractical to use.

It works optimum and useful for few users and channels. Therefore we break network into small clusters which are optimised first then the nodes above in bottom up approach.