Chord is a protocol and algorithm for a peer to peer distributed hash table which stores key value pair by assigning keys to different computers; a node will store the values for all the keys for which it is responsible (https://en.wikipedia.org/wiki/Chord_(peer-to-peer)). The key of nodes will be distributed uniformly on the Chord ring. Each node keeps a finger table that records the key and address of the nodes which has the distance 2^n from it. For each lookup, the adjacent node which have the longest distance on the Chord ring will be selected. Each Internet hop will reduce the search space by half, like the bisection search. This greedy approach is efficient and can reduce the complexity of lookup algorithm to O(log(N)). But this approach has not considered the latency of underlying network, which also has a significant influence on the lookup latency. For the original server selection approach used in the Chord layer of peer to peer system, each node use the latency feed back from RPC to estimate the statistics of underlying network. And then used the information to decide the best choice for next Internet hop. But this approach assumes that the local latency pattern can reflect the global distribution. It will not make accurate prediction when the distribution of latency is not uniform.

In this project, a reinforcement learning approach has been developed to address this problem. a Chord ring simulator has been developed which can be used to study the lookup algorithm. The cooperative q learning, together with the greedy algorithm and server selection approach has been implemented and the result has been compared. The cooperative q learning algorithm can self improving and finally get a better result compared with other approaches.

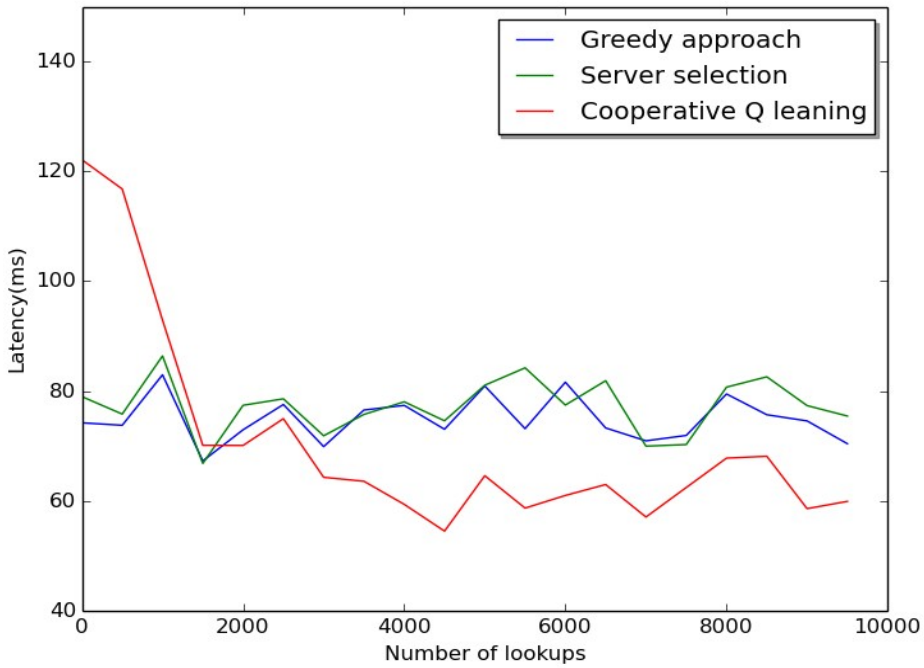

Fig 1, comparison of the lookup latency between different approach