*In your report, mention what you see in the agent’s behavior. Does it eventually make it to the target location?*

When choosing the action randomly, 47 out of 100 trails reach destination within dealine+100 steps. But most of the agent reach destination beyond deadline limitation.

*Justify why you picked these set of states, and how they model the agent and its environment.*

There are many ways to choose the state and I use input and next\_waypoint as the state. Inputs including the traffic lights and oncoming cars indicates the current traffic conditions. And next\_waypoint gives information of the correct direction and helps the agent to learn more effectively.

*Q-learning: What changes do you notice in the agent’s behavior?*

*Report what changes you made to your basic implementation of Q-learning to achieve the final version of the agent. How well does it perform?*

When using random actions, only 47 episodes arrive at the destination within time limit but after applying Q-learning, all 100 episodes arrive at the destination. I run the agent for 5 times, and the agent learn how to obey traffic rules after about 70 trails because it gets no negative reward after 70th trail. Although, it may choose the suboptimal action, which can be proved by the 0 reward, the agent will not crash in car accident and thus reach the destination more quickly.

*Does your agent get close to finding an optimal policy, i.e. reach the destination in the minimum possible time and not incur any penalties?*

In order to evaluate the effectiveness of the model, I print out total steps to reach the destination. However, it is unreasonable to use total steps as a bench mark because if the Manhattan Distance is large, total step must be large. Therefore, I define total step/Manhattan Distance as step-efficiency to assess effectiveness.

Average reward = /100 Average step-efficiency =

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Agent | Epison | Gamma | Alpha | Average reward | Average step-efficiency |
| Agent1 | 0.02 | 0.8 | 0.7 | 1.77 | 4.02 |
| Agent2 | 0.05 | 0.8 | 0.7 | 1.82 | 3.53 |
| Agent3 | 0.02 | 0.5 | 0.5 | 1.92 | 4.01 |
| Agent4 | 0.05 | 0.5 | 0.5 | 1.97 | 2.56 |
| Agent5 | 0.05 | 0.7 | 0.7 | 2.05 | 3.08 |

According to the above table, epison=0.05, gamma=0.7 and alpha=0.7 agent has the best results. I am going to evaluate whether this agent learn the best policy.

In my opinion, if the agent has learned the best policy, it chooses the same action as GPS chooses and it will get a reward 2.0, and thus the average reward should be larger than 2.0. Besides, the agent must reach destination within deadline and it means average step-efficiency is less than 5.0.

Obviously, agent5 has reached the optimal policy.