Reinforcement Learning On Intersection Management

B08902116, 賴世宗
B08902142, 蕭啓湘

January 8, 2022

Contents

1 Abstract

In Intelligent Vehicle, the intersection management is a significant part. In intersection management, there is several methods depend on the type of vehicles; connected vehicle, autonomous vehicle;, which is the controlling device; vehicles or traffic signals. In this project, we propose the reinforcement learning method on controlling vehicle pass through the intersection frame by frame which can be used on the connected vehicle and traffic signal combining together.

After the experiments, we will evaluate the performance and explain the result, also there will be some improvement suggestions following.

2 Introduction

The field of Intelligent Vehicle has involved the intersection management in the point that it has played an important role effect to the vehicle system. In traditional, traffic signal controllers are using fixed hand-crafted control, which do not consider the current intersection's condition (e.g. fixed cycle-based methods) which is restricted from current human-based vehicle but in the future, as the intelligent vehicle technology being more and more capable, the controlling and cooperating between intelligent vehicle and traffic signal are more feasible. The heuristic algorithms and models are applicable on the vehicle.

3 Dataset

Randomly generates cars and cars' direction by setting variables of total cars, congestion rate(probability of cars generated in each time frame)

4 Model Architecture

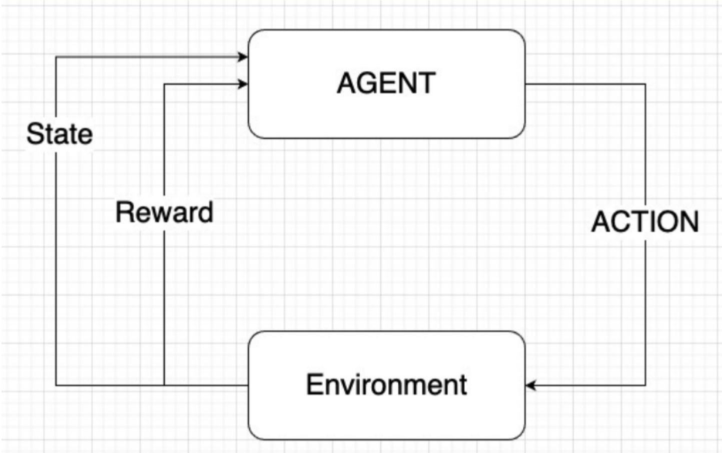

Figure 1: Reinforcement Learning Model

4.1 Environment

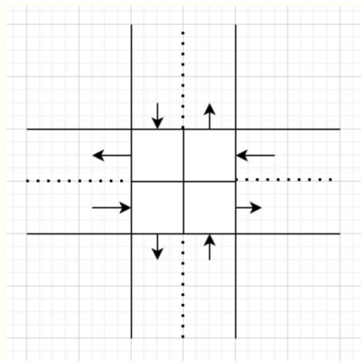

Figure 2: Environment

Environment is the intersection with 4 directions, one lane each direction and 4 conflict zones divided, included cars waiting list in each directions. The basic intersection is designed as a way to test the feasible of the implementation.

State of the environment sent to the agent is the current condition of each conflict zones and car's waiting list in each lanes.

4.2 Agent

Receiving the status of the environment and giving out the possibility of the actions for each cars by sending the state to the neural network.

4.3 Reward

Rewards for action of the agent are cars reaching the destination, cars taking right action and the penalty are cars do not taking action when the action is available, the actions cause deadlock and cars crashing.

5 Our Training Method

5.1 Flowchart Diagram

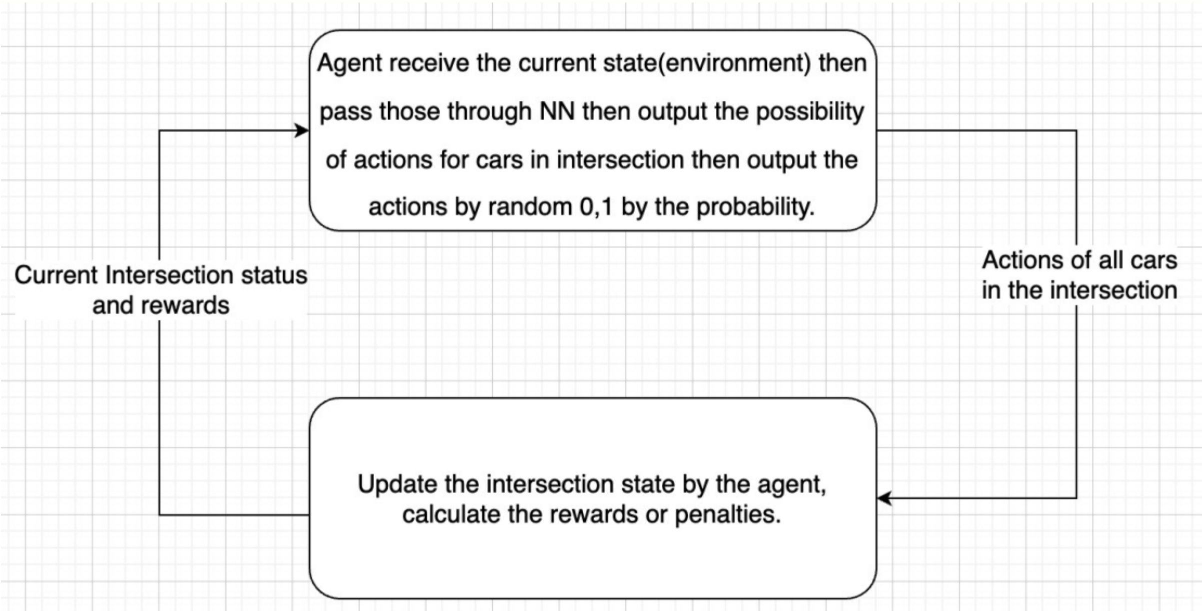

Figure 3: Flowchart

6 Experiments

train env	(6,0.75)	(6,0.25)	(10,0.75)	(10,0.25)	(40,0.75)	(40,0.25)
test env						
(40,0.25)	65.27	71	102.13	75.87	91.13	75.53
(40,0.75)	59.13	69.73	101.07	72.4	86	74.2
(10,0.25)	25.67	24.2	31.07	30.33	32.4	28.27
(10,0.75)	21.93	23.87	32.33	24.33	30.8	23.33
(100,0.25)	141.2	156.53	241.93	176.93	205.13	165.27
(100,0.75)	130.47	155.13	233.2	175.73	200.87	160.83

Figure 4: Experiment Result

The column label is training environment , The row label is test environment, and the number in each grid is the average timeframe used in test environment.

7 Evaluation Method

We evaluate our model by total timeframes needed to solve the traffic flow in each number of cars and congestion rates.

8 Performance

By the experiment result, it is seen that the model result is far from the optimal solution. However, this model is can handle the traffic flow correctly.

9 Future Work

1. Increase the intersection variety (e.g. number of lanes)
2. Increase the performance of the model
3. Take the car's performance into account(maybe some cars can move 3 grid in a timeframe but others can only move 1)
4. Reduce and deal with crash and idle

10 Conclusion

In our experiment, we can find that no matter what environment in test case, using (low number of cars , high congestion rate) environment in training can lead to a better result. According to this result , we speculate that using such environment has balance situation in traffic flow, like entrance period(there is no car in the intersection and cars are ready to enter), congestion period(there are some cars in the intersection and lots of car want to enter), end period(there are some cars in the intersection and no car need to enter), and etc.

The reinforcement learning model has high randomness. Even with the best model , there is still a chance lead to crash or idle . How to deal with this is also an important topic to improve the model.

11 Appendix

11.1 Our code

Github

11.2 Work Contribution

1. 賴世宗：Model Environment, Agent, Partial Training, Presentation making, Report
2. 蕭啓湘：Model Training, Project Presentation, Report

References

Code workflow based on Machine Learning (Spring 2020) class.