

Using AI and Machine Learning Techniques for Traffic Signal Control Management- Review

Professor Sunil Ghane, Vikram Patel, Kumaresan Mudliar, Abhishek Naik
Sardar Patel Institute of Technology, Mumbai
Mumbai, India

Abstract—Traffic congestion has been a problem affecting various metropolitan areas. This can result in various problems such as untimely delays to commuters, increase in carbon dioxide emissions and increase in travel time. The current traffic control systems fail to consider the volume of traffic in a particular junction i.e. the duration of the green or red signal is fixed and does not depend on the density of vehicles. Here we are reviewing the paper which provides the solution to the current traffic control systems with the help of neural networks. This paper also compares the solution with the current traffic control systems as well as another solution which uses the concept of q learning.

Keywords— Component; Formatting; Style; Styling; Insert

I. INTRODUCTION

Traffic congestion is a major problem in various cities. Due to increase in urbanization and number of vehicles on roads in recent years, there is a need for an efficient traffic control systems to avoid congestion. The congestion can result in unexpected delays to people travelling by road and can result in inability of emergency services such as ambulance to reach the required destination on time. For this purpose, traffic signal control becomes a major factor in controlling traffic in an efficient manner and avoid problems due to traffic congestion. The traditional signal system is a fixed time system. Here the signal displays the green signal for a fixed duration of time. Some other systems allow us to set time for the duration of green signal depending on the time of the day. So one can set more duration for the green signal during peak hours and less duration during the late hours when the roads are not busy. However the traditional systems do not consider the factor of number of vehicles on the road at the given time. The disadvantage of such systems is that this system cannot customize the duration of green signal based on the current traffic conditions. This may result in traffic congestion problems on junctions where the duration of green signal is not enough to clear the lanes with heavy traffic. The models will help us to decide which lane needs more duration of green signal on the basis of number of vehicles in a lane rather than fixed duration. This model also helps reduce waste of time on roads where there are less vehicles and instead utilize that time for the other busy roads.

II. THE PROPOSED SOLUTION

Given a junction our aim is to determine the best possible duration of the cycle and each phase in it based on the current

incoming traffic in each of the four lanes. The duration of each phase should be flexible so that congestion does not occur in any lane. Even the vehicle density of each road will be calculated and depending upon the density of each road the signal will be programmed so that in high density roads the signal will display green for a greater duration of time and with this process the traffic can be controlled dynamically.

A. Q Learning Method

Q Learning [1] is one of the techniques which employs reinforcement learning. Reinforcement learning is a type of learning in which agents in a particular state take an action on the basis of the information received from the environment. Here the information acts as a feedback for the software agents. This feedback can be a reward or a punishment on the basis of which these agents take actions in the environment. Q-Learning is model free technique, which means that this method can be used without any model of the present environment. This method uses a Markov decision process. Here an agent in a state 's' performs action 'a' and on the basis of action deserves a reward 'r'. Depending on this reward, the agent decides whether the particular action has to be taken or not. The formula for the given process is

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha(r_t + \gamma(\max(Q(s_{t+1}, a))))$$

Where s. represents the current state (at time t) a. represents the action taken at time t, represents the learning rate and represents the discount factor.

The learning rate represents the degree of tendency to move to another state. For example if the learning rate is 0, the agent will not move to any new state. So more the , the agent will be more accustomed to moving to new states. represents the discount factor. The importance of future rewards is determined by the discount factor.

1. Set initial value of $Q(s, a)$ at random
2. For one episode repeat from steps 3 to 8
3. Select an initial state ' s_0 '
4. For each step in an episode repeat from steps 5 to 8 until the desired state is reached.
5. Select an action 'a' from a set of actions on the basis of a policy used by Q (ϵ -greedy)
6. Note result r and the new state s_{t+1} after performing the action a_t
7. $Q(s_t, a_t) = Q(s_t, a_t) + \alpha(r_t + \gamma(\max(Q(s_{t+1}, a))))$
8. $S_t = S_{t+1}$

Traffic Model

In our current traffic signal problem, our agent is nothing but the traffic signal at an intersection. For our current scenario we select the states identified by the agent. The states are the average length of the queue of vehicles in the approaching links in a fixed cycle. The number of states is the permutation of the number of approaches. Here in our case at the junction there are 4 approaching links, so our total number of states will be $4! = 24$. Each state here represents a comparison of the queue length of the approaching links. For example consider l_i to be the queue length of the i^{th} link then $l_1 \geq l_2 \geq l_3 \geq l_4$ represents a state where link 1 has the longest queue length as compared to the other links while link 4 is the shortest. In all there are 24 such states as represented by the table below.

State	Link Order	State	Link Order
1	$l_1 \geq l_2 \geq l_3 \geq l_4$	13	$l_1 \geq l_2 \geq l_4 \geq l_3$
2	$l_1 \geq l_2 \geq l_4 \geq l_3$	14	$l_1 \geq l_3 \geq l_2 \geq l_4$
3	$l_1 \geq l_3 \geq l_2 \geq l_4$	15	$l_1 \geq l_3 \geq l_4 \geq l_2$
4	$l_1 \geq l_3 \geq l_4 \geq l_2$	16	$l_1 \geq l_4 \geq l_3 \geq l_2$
5	$l_1 \geq l_4 \geq l_2 \geq l_3$	17	$l_1 \geq l_4 \geq l_3 \geq l_2$
6	$l_1 \geq l_4 \geq l_3 \geq l_2$	18	$l_1 \geq l_4 \geq l_2 \geq l_3$
7	$l_2 \geq l_1 \geq l_3 \geq l_4$	19	$l_2 \geq l_1 \geq l_4 \geq l_3$
8	$l_2 \geq l_1 \geq l_4 \geq l_3$	20	$l_2 \geq l_1 \geq l_3 \geq l_4$
9	$l_2 \geq l_1 \geq l_3 \geq l_4$	21	$l_2 \geq l_1 \geq l_4 \geq l_3$
10	$l_2 \geq l_1 \geq l_4 \geq l_3$	22	$l_2 \geq l_3 \geq l_1 \geq l_4$
11	$l_2 \geq l_3 \geq l_1 \geq l_4$	23	$l_2 \geq l_3 \geq l_4 \geq l_1$
12	$l_2 \geq l_3 \geq l_4 \geq l_1$	24	$l_2 \geq l_4 \geq l_3 \geq l_1$

Fig1 State space representation

The action that the agent will take in this case is the management of duration of green signal in each phase of the cycle. Here based on each state the agent ie. the traffic signal at the intersection will determine what should be the duration of the green signal at each phase (at one approaching link) Based on the reward factor the duration may increase or decrease to reduce the delay.

We consider the cycle time at each junction to be fixed. Cycle time is the total duration required to change all 4 signals in a junction and return to the starting signal. $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 1$. Here each phase split refers to the duration of green signal for each approaching link. We denote the total time of the junction as T .

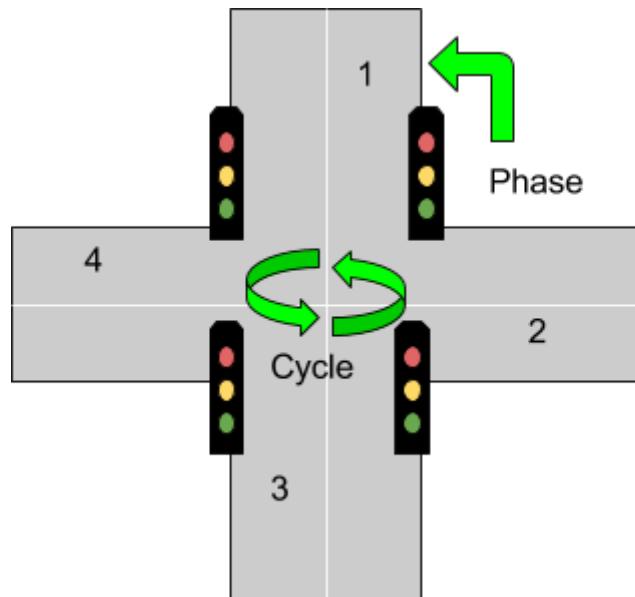


Fig 2: Representation of phase time and cycle time

Here N refers to the number of phases in a cycle. Here we ensure that each phase has a minimum duration of green signal. We denote this by T_{min} . Also we can allow a small extension time for each phase ie more duration of green time along with the fixed minimum. Let A_{xt} denote the extra duration and N_a denote the number of extensions. So the total cycle time is given by

$$T = N * T_{min} + N_a * A_{xt}$$

Here the reward of each stage is inversely proportional to average queue length of approaching link. Hence the signal will approach those stages which in turn reduce the queue length. The reward in here is normalized and it is restricted between 0 and 1.

B. Neural Network

A Neural Network [2,3] is a paradigm which processes information. Its working is formulated in the manner in which biological neural networks work. It is a series of interconnected nodes which when given a series of input gives a particular required output. It has an excellent capability to interpret meaning and interdependency from a given set of data which is difficult to be noticed by humans or other computer techniques. Neural Networks have the ability to learn how to do tasks only on the basis of information provided by the data. Neural Networks can also determine representation and organization of data during learning phase.

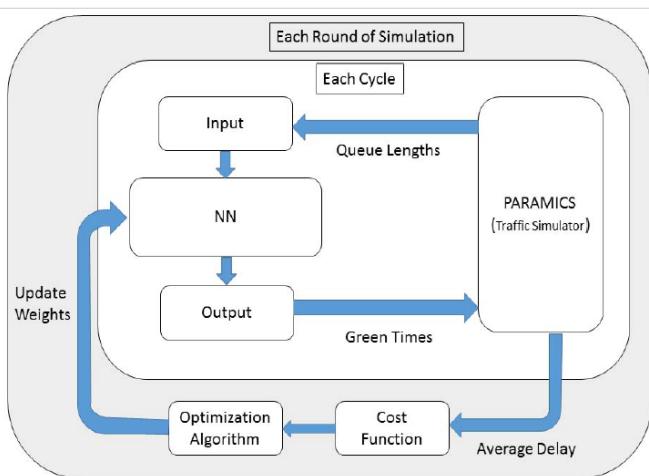


Fig 3: Flow diagram depicting training of neural network.

Here PARAMICS, a traffic simulator has been used to simulate a real time network traffic to provide data to the neural network. The feed forward type of ANN has been used to provide duration of green signal. Here the input layer has four neurons, ten hidden layer neurons and four output layer neurons. The optimization method used for the ANN is simulated annealing.

Algorithm:

1. Queue length of the approaching links is given as input to the neural network by the simulator.
2. The Neural Network processes the input and provides green time for each link at the output.
3. This output is given as input to the simulator which in turn provides the delay time occurring as the cost function
4. Based on the cost function and Simulated Annealing methods, new weights for the ANN are generated and the updated weight are used for the next input.

III. COMPARISON

Here we have used PARAMICS to calculate cumulative delay for each hour upto 5 hours for 5000 vehicles and checked the average delay time for each hour.

Hour(s)	Fixed time method	Q-learning Method	Neural Network
1	5317	5223	2764
2	11193	10025	5228
3	17644	15354	8334
4	21135	20637	10785
5	26417	25917	14275

Fig 4: Observation and comparison

Here we can see that neural networks provided the highest efficiency in terms of less delay time. Here we also observe that the difference between delay times of traditional networks and Q-learning methods is quite less. So even though Q-learning methods are better than traditional networks, they are less efficient as compared to the neural network method.

Another problem with Q-learning methods is that it only considers traffic in one junction and not in other junctions and as a result does not consider the traffic in overall network. Hence there might arise a situation due to which the signal may reduce traffic in one junction but result in congestion in other.

IV. OBSERVATIONS

Even though Q-learning methods are better than traditional networks, they are less efficient as compared to the neural network method.

Another problem with Q-learning methods is that it only considers traffic in one junction and not in other junctions and as a result does not consider the traffic in overall network. Hence there might arise a situation due to which the signal may reduce traffic in one junction but result in congestion in other.

V. FUTURE SCOPE

The current systems can be modified by considering the total congestion in neighboring systems while calculating the reward or cost function. In this way the model chooses the next step by considering not only the traffic in approaching links but also the neighboring junctions.

REFERENCES

- [1] T. Parisini, M. Sanguineti, and R. Zoppoli, "Nonlinear stabilization by receding-horizon neural regulators," *Int. J. Control.*, vol. 70, no. 3, pp. 341–362, 1998.
- [2] P. B. Wolshon and W. C. Taylor, "Analysis of intersection delay underreal-time adaptive signal control," *Transp. Res., Part C Emerg. Technol.*, vol. 7C, no. 1, pp. 53–72, Feb. 1999.
- [3] W. Wei and Y. Zhang, "FL-FN based traffic signal control," in *Proc.FUZZ-IEEE*, May 12–17, 2002, vol. 1, pp. 296–300.