Environmental Adaptive Urban Traffic Signal Control Based on Reinforcement Learning Algorithm

Yanming Feng, Yongrong Wu*
School of Information and Communication, Guilin University of Electronics Technology, Guilin 541004, China

*Corresponding author:

Abstract. Urban traffic signal control is an important part of the construction of intelligent regional traffic. Aiming at the problem of the optimal control strategy in urban traffic signals, this paper proposes environmental adaptive urban traffic signal control based on reinforcement learning algorithms. Through the continuous perception of the traffic environment, the position and speed of the vehicles in different environments are expressed in a matrix, and the parameters are continuously iteratively optimized through the reinforcement learning method to optimize the objective function (the vehicle that can pass the most in a limited time) to achieve the purpose of effective vehicle control. According to the traffic simulation software Vissim, it can be known that the algorithm proposed in this paper performs better in terms of average waiting queue length and global average speed compared with other algorithms, and the deep learning algorithm is significantly better than other algorithms in terms of stability. The average speed of the deep learning algorithm is increased by 9% compared with the baseline, and the average waiting queue length is reduced by 13.4% compared with the baseline. The experiments studied this time are sufficient to prove that the algorithm in this paper can adapt to the dynamically changing and complex urban traffic environment and has great research value.

Keywords: Traffic signal control; environment adaptive; reinforcement learning.

1. Introduction to deep reinforcement learning
The traditional control of single point signal in urban traffic usually relies on the managers to observe and debug signal control facilities and crossing signals. The increasing complexity of traffic signal control brings challenges to the operation and coordination of urban traffic flow. At present, the control and optimization of urban traffic signals usually use deep learning algorithms to collect, mine and analyze the massive data resources of the traffic system, complete the design of adaptive signal timing and scheduling phases of urban traffic intersections, and improve the prediction, decision-making and control of traffic signals in different regions [1].

Reinforcement learning belongs to interactive learning, a type of machine learning, which can interact with the environment in real time, so as to clarify the goal and achieve the optimization of the goal. The basic framework of reinforcement learning is shown in Fig. 1. The reward value is a feedback signal that occurs after the agent performs an action. The agent observes the environment
state $S_t$ at each time step, formulates an action of $a_t$, receives the reward value $r_t$, and updates the environment state $S_{t+1}$ [2]. After a period of interaction, the agent obtains a series of actions, states and rewards. The behavioral basis of the agent is the strategy $\pi$, which belongs to a kind of mapping that occurs from the state of the environment to the actions of the learning agent. $G_t$ is the cumulative reward value between the start of the learning agent and the time $t$. The formula for cumulative reward $G_t$ is as follows:

$$G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$$ (1)

In formula (1), $\gamma$ is the discount factor, and $0 < \gamma \leq 1$. Finding the optimal strategy $\pi^*$ is the main goal of an agent, and the goal can be regarded as the expectation of maximizing the accumulated reward value, expressed by the number of rewards.

![Figure 1. The agent-environment interaction](image)

When the number of states in the environment exceeds the capacity of modern computers, designers combine Q-learning and CNN in the reinforcement learning algorithm to propose a deep reinforcement learning control algorithm (DQN) based on environmental adaptive urban traffic signals [3]. In this way, the DQN algorithm can be used in traffic signal control problems. This article mainly analyzes the application of DQN algorithm in urban traffic signal control. In the DQN algorithm, the agent interacts with the environment and receives data, which will be used in the learning process of the Q network. The agent explores the environment and builds a full picture of conversion and action output.

2. Establishment of DQN Model of Deep Reinforcement Learning Based on Adaptive Environment

In this section, we analyze the various parts that need to be concerned in building DQN (NIPS 2015) model in urban traffic signal control.

2.1. State Representation Analysis

The current state can be represented by the vehicle position information and local average speed in the traffic network, and can be represented by two matrices. In the state representation method, the length of the formal cell in which the lane is divided into cells is certain. Each cell has a one-to-one correspondence with an element in the matrix, while the average vehicle speed and the number of vehicles can be represented by the cell corresponding to each element in the matrix. Under the state representation method, the main information can be extracted from the traffic network state, and the
agent can make decision analysis based on the extracted information [4]. The representation method of environmental state is shown in Fig. 2.

2.2. Traffic Signal Control Agent Analysis
The main part of the traffic signal control agent is analyzed through the deep neural network. The traffic environment state is the input part of the deep neural network, and the state action value Q of all legal actions is the output part of the neural network. The agent can refer to the value function when choosing the optimal action. Given any environment state, only one forward propagation algorithm is needed to get the state action value of all actions, which is the advantage of deep neural network structure. The convolutional layer, pooling layer, and fully connected layer are the main structures of the agent. The convolutional layer uses a 4×4 convolution kernel with a stride of 1, and the pooling layer uses a stride of 1, with a size of the 3×3 convolution kernel and the fully connected layer select a 3×3 convolution kernel with a stride of 1 and a size of 1. ReLU function as hidden layer activation function [5].

\[
\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 2 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 2 & 0 \\
2 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{pmatrix}
\]

\[
\begin{pmatrix}
35 & 0 & 0 & 0 & 0 & 0 & 0.2 \\
0 & 0 & 10 & 0 & 0 & 0 & 11 \\
0 & 0 & 0 & 30 & 0 & 22 & 0 \\
2.2 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{pmatrix}
\]

\text{Position Matrix}

\text{Velocity Matrix}

\begin{figure}
\centering
\includegraphics[width=0.5\textwidth]{environmental_state_representation.png}
\caption{Environmental state representation}
\end{figure}

2.3. Reward Value Function Analysis
The agent executes the \(a_t\) action at the time step \(t\), and then can obtain the feedback of the environment for this action, which is the reward value \(r_t\). The formula for reward value \(r_t\) is as follows:

\[
r_t = a(V_{t+1} - V_t)
\]  \hspace{1cm} (2)

In formula (2), \(r_t\) is the difference between the global average speed of adjacent time steps. At time \(t\), the average speed of all vehicles in the traffic network is \(V_t\), and the global average speed at the next time can be regarded as \(V_{t+1}\). As a constant quantity, \(a\) can determine the reward value in a range without large fluctuations. In this study, \(a\) is set to 0.1 under multiple experiments.

2.4. D. Learning Algorithms Analysis
In order to train the agent, it is necessary to ensure that the corresponding target value is output every time. However, in practice, it’s more difficult to obtain the target value. Based on this, it is necessary to select an approximate target value \(\text{TargetQ}\) [6]. The formula of approximate target value \(\text{TargetQ}\) is as follows:

\[
\text{TargetQ} = r + \gamma \max_{a'} Q(s', a'; \theta^-)
\]  \hspace{1cm} (3)
In formula (3), the state action value $Q$ is the output content of the neural network in each iteration. $\theta^-$ is the network parameter for calculating the target value. $\theta^+$ is the network parameter for calculating the target value.

If the approximate standard state action function is represented by a nonlinear function approximator, the reinforcement learning will be unstable due to the correlation between the data. In order to solve the above problems, the relevant designers proposed to use the classic playback mechanism to solve the problem. The replay memory unit $D_t$ stores the data of the interaction between the agent and the environment. During the training of the agent, small batches of data are randomly selected from the memory unit. Compared with the data collected by the Q-learning algorithm, it can be used repeatedly to update the network weights, which can guarantee the utilization of the data. Randomized samples can break the correlation and improve the learning efficiency [7].

3. Experiment Analysis
In order to verify the effectiveness of the urban traffic control algorithm proposed this time, a desktop with Win10 OS was used to run the experiment. The scheduling algorithm is implemented by Python. Then evaluate scheduling policies under different parameter settings. The simulation experiment based on the traffic control algorithm of the Vissim platform can simulate the authenticity of the urban traffic scene, thereby obtaining a highly reliable urban traffic modeling.

3.1. Experimental Setup
Choose a 2×2 grid-shaped urban road network as the simulation task scene. The structure of all intersections is the same, and the interval between intersections is 400m. The flow of traffic entering each intersection is also the same. If 600veh/h is defined as the initial traffic flow, if the signal controller does not have a full red-light buffer time, it can directly enter the next phase after finishing one phase. During the agent training, each scene including 150-time steps, which is the same as the simulation time of 25 minutes. The simulation tasks scene shown in Fig. 3.

![Figure 3. 2×2 grid-shaped urban road network](image)

3.2. Optimization Hyperparameter
In the initial training stage of the agent, the relevant parameters can be set as follows: learning rate $10^{-4}$, buffer pool size 3000, discount factor 0.9, initial epsilon 0.6, decay rate 0.99, and minimum epsilon 0.01. After the agent needs to reset the testing phase after the end of the training Vissim test environment, test time is 25min.

The evaluation indicators selected this time are the global average speed and the average waiting queue length. The global average speed indicator can describe the cumulative reward value of the transportation network state. If the decision made continuously by the agent is wrong after the 50th time step, the traffic network situation becomes worse at this time. At this moment, the optimization of hyperparameters is needed to improve the performance of the agent. In the experiment, in order to
improve the performance of the agent, it is necessary to change the traffic flow on the basis of the agent. After repeated experiments, the traffic flow range will increase, the complexity of the traffic state will increase significantly, and the training difficulty of the agent will increase. Eventually it will affect the performance of the agent. On the basis of the original test, a low-traffic scenario test task on the agent is added. This high-traffic environment training can be realized in a low-traffic environment [8-9].

3.3. Evaluation of Urban Traffic Signal Control Parameters
Traffic signal control aims to maximize the traffic benefits of single-point intersections or road networks by setting reasonable signal timing schemes. When evaluating the traffic benefits of a traffic system network, various factors must be considered comprehensively. The traffic designer can choose different evaluation indicators according to actual needs, and can select single or multiple indicators to construct the system's indicator evaluation function. The general evaluation indicators functions include delay time, saturation, queue length and capacity. We use the average delay of vehicles as the evaluation index, and the delay time is introduced as: The difference between the time required for a motor vehicle to travel on the road without any traffic interference, traffic control facilities and management and the actual travel time. According to the reasons, the delay time can be divided into: fixed delay, operation delay, parking delay, travel time delay, queuing delay, approach delay and control delay, etc. The total delay is the difference between the actual travel time and the reference travel time. The reference travel time is the travel time spent under ideal conditions with no traffic control, no geometric delays, no traffic incidents, and other vehicles.

3.4. Analysis of Experimental Results
From Table 1, under smooth flow, compared with two-level fuzzy control and timing control, the average vehicle delay of the combined optimization control model is reduced by 6% and 37%, respectively. Under Steady flow, it is compared with timing control, fuzzy control and chaos. Compared with the two-level fuzzy control before genetic algorithm optimization, the average vehicle delay of the combined optimization control model is reduced by 39%, 20%, and 8% respectively; under congested traffic, it is compared with the two levels before optimization of timing control, fuzzy control and chaotic genetic algorithm. Compared with the first-level fuzzy control, the average vehicle delay of the combined optimization control model is reduced by 33%, 23%, and 12%, respectively. Under the congestion flow, the combined optimization control model does not change significantly compared with other control methods.

| Table 1. Average delay of different controls under traffic flow |
|---------------------------------------------------------------|
|                  | Timing control | Fuzzy control | Two-level fuzzy control | Combinatorial optimization control |
| Smooth flow      | 20.6           | 13.1          | 13.9                     | 13.1                                    |
| Steady flow      | 34.7           | 26.8          | 23.1                     | 21.2                                    |
| Crowded stream   | 57.3           | 49.5          | 43.4                     | 38.1                                    |
| Clogging flow    | 76.7           | 73.6          | 73.2                     | 73.6                                    |

Taking the average delay of the vehicle as the evaluation index, when the entrance vehicle at the intersection is 2130pct/h, the algorithm performance of the comparison method and our algorithm is shown in Table 2. It shows that the search efficiency of our algorithm is higher, and the average delay of vehicles is smaller.
Table 2. Algorithm performance comparison

|                                | contrast method | our method |
|--------------------------------|-----------------|------------|
| Number of trials               | 10              | 10         |
| Number of evolutions           | 100             | 100        |
| The average of the optimal solution | 22.1          | 19.8       |
| The optimal solution occurrences | 53             | 67         |

4. Conclusions
In summary, in view of the continuous increase of urban traffic pressure, according to the optimal control strategy in urban traffic signals, an enhanced learning control algorithm for urban traffic signals based on environmental adaptation is proposed. Through continuous perception of the traffic state and mining hidden patterns, the optimal control strategy is found. Experiments show that urban traffic can be effectively controlled under this method, and the traffic efficiency has been significantly improved.

Acknowledgements
National Natural Science Foundation of China (61962016).

References
[1] Lu Shoufeng and L. Ximin, "Based on Hybrid Genetic Algorithm and Cellular Automata Combined Traffic Signal Control and Route Guidance," 2007 Chinese Control Conference, Hunan, 2007:53-57.
[2] Sutton, R, Barto, A. Reinforcement Learning: An Introduction [M]. MIT Press, 1998.
[3] Amir T, Alireza Z S. Directional channel modelling for millimetre wave communications in urban areas [J]. IET Communications, 2018, 12(6):656-664.
[4] Morrison A, Manresa-Yee C, Knoche H, et al. Vibrotactile and vibroacoustic communications: pairs in interaction and play—an interactive structure and bodies in an urban environment [J]. Universal Access in the Information Society, 2017.
[5] Samuylov A, Ometov A, Begishev V, et al. Analytical performance estimation of network-assisted D2D communications in urban scenarios with rectangular cells [J]. Transactions on Emerging Telecommunications Technologies, 2017, 28(2): 1-15.
[6] Melrose Roderick, James MacGlashan, Stefanie Tellex, “Implementing the Deep Q-Network”, ArXiv e-prints, 2017.
[7] A. R. Fayjie, S. Hossain, D. Oualid and D. Lee, "Driverless Car: Autonomous Driving Using Deep Reinforcement Learning in Urban Environment," 2018 15th International Conference on Ubiquitous Robots (UR), Honolulu, HI, 2018:896-901.
[8] Unterhuber P, Sand S, Fiebig U C, et al. Path Loss Models for Train-to-Train Communications in Typical High Speed Railway Environments [J]. IET Microwaves Antennas & Propagation, 2018, 12(4): 492-500.
[9] Fadda M, Murroni M, Popescu V. Interference Issues for VANET Communications in the TVWS in Urban Environments [J]. IEEE Transactions on Vehicular Technology, 2016, 65(7): 4952-4958.