



# A Reinforcement Learning-Based Controller Designed for Intersection Signal Suffering from Information Attack

Longsheng Ye<sup>1</sup>, Kai Gao<sup>1,2</sup>(✉), Shuo Huang<sup>1</sup>, Hao Huang<sup>1</sup>, and Ronghua Du<sup>1,2</sup>

<sup>1</sup> College of Automotive and Mechanical Engineering, Changsha University of Science and Technology, Changsha 410114, China

kai\_g@csust.edu.cn

<sup>2</sup> Hunan Key Laboratory of Smart Roadway and Cooperative Vehicle-Infrastructure Systems, Changsha University of Science and Technology, Changsha 410114, China

**Abstract.** With the rapid development of smart technology and wireless communication technology, Intelligent Transportation System (ITS) is considered as an effective way to solve the traffic congestion problem. ITS is able to collect real-time road vehicle information through sensors such as networked vehicles (CV) and cameras, and through real-time interaction of information, signals can more intelligently implement adaptive signal adjustment, which can effectively reduce vehicle delays and traffic congestion. However, this connectivity also poses new challenges in terms of being affected by malicious attacks that affect traffic safety and efficiency. Reinforcement learning is considered as the future trend of control algorithms for intelligent transportation systems. In this paper, we design reinforcement learning intelligent control algorithms to control the intersection signal imposed by malicious attacks. The results show that the reinforcement learning-based signal control model can reduce vehicle delay and queue length by 22% and 23% relative to timing control. Meanwhile, the intensity learning is a model-free control method, which makes it impossible for attackers to target flaws in specific control logic and evaluate the impact of information attacks more effectively. Designing a coordinated state tampering attack between different lanes, the results show that the impact is greatest when the attacked states are in the same phase.

**Keywords:** intelligent transportation system · reinforcement learning · information attack

## 1 Introduction

With the high-speed development of society and the increasing number of cars, traffic congestion has increasingly become an important factor limiting urban development. And an excellent signal control system is of great significance for improving traffic efficiency [1]. A key point of traffic signal optimization is traffic status sensing, and in the

This work was supported in part by the Natural Science Foundation of Hunan under Grant 2021JJ40575, in part by The Open Fund of Hunan Key Laboratory of Smart Roadway and Cooperative Vehicle-Infrastructure Systems (Changsha University of Science & Technology) under grant kfj190701.

past, intersections were used to obtain real-time traffic status through induction coils and cameras, but these devices have the disadvantages of high installation and maintenance costs and short service life [2, 3]. Thanks to the development of vehicle networking and Vehicle-to-Everything (V2X) communication, a Connected Vehicle (CV) driving on the road can act as a powerful motion detector to provide high-quality and inexpensive traffic status input to signal intersections [4–6]. With the further development of communication and artificial intelligence technologies, Intelligent Transportation System (ITS), which combines the concepts of vehicle networking and intelligent control, is gradually becoming the future direction of transportation systems, making it possible to improve traffic efficiency at urban intersections on a large scale and in all directions [7, 8].

Combined with a data-driven control approach, the efficiency of traffic junction passage is effectively improved. Reinforcement learning (RL) is an unsupervised machine learning algorithm. Unlike supervised learning approaches that require pre-labeling, reinforcement learning models learn entirely through interaction with the environment until the best policy is learned to guide the action, which is a very advanced adaptive control method [9]. Data-driven reinforcement learning control enables intersection signal control to be fully integrated with the real-time characteristics of traffic flow, which is important for improving traffic flow efficiency [10]. Traditional reinforcement learning is suitable for simple models such as segmented constant tables and linear regression, leading to limited scalability or optimality in practice. Deep Neural Network (DNN), which has the ability to learn complex tasks [11, 12], combined with reinforcement learning solves this problem and brings a new direction to intersection adaptive control. To improve the stability of the algorithm, empirical replay and target network mechanisms are used. Simulation results show that the algorithm reduces vehicle delays by 47% and 86%, respectively, compared to two other popular traffic signal control algorithms, the longest queue priority algorithm and the fixed time control algorithm [12]. Liang et al. use a rasterized state input representation to quantify the traffic flow information by dividing the intersection into occupancy raster maps. Multiple states such as vehicle position and speed are fused and different green light times are set as action spaces. A convolutional network is used to approximate the Q-network and mechanisms such as empirical replay and target networks are employed to stabilize the Q-network [13].

This paper focuses on designing a reinforcement learning based control model for intersection signal controller suffering from information attack. There are two main reasons, firstly to evaluate how much improvement the reinforcement learning model can bring to the efficiency of the intersection communication. Secondly due to the model-free nature of reinforcement learning, the evaluation of the impact of being attacked is more convincing, this is because the attacker cannot set a specific targeting strategy based on the flaws of the model.

The main contributions of this paper are summarized as follows:

1. A reinforcement learning-based signal control model is designed that uses a deep neural network to approximate the maximum cumulative return and an empirical replay mechanism to stabilize the network.

2. Performing a signaling attack on the reinforcement learning control model. The interference of the information attacks is removed by using the model-free property of reinforcement learning.
3. The state of the controller is tampered with to explore the best attack strategy.

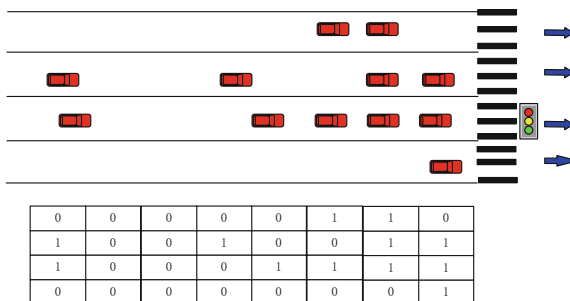
## 2 Problem Description

### 2.1 Application Scenarios

To solve traffic control problems, intelligent transportation systems (ITS) have been created by integrating advanced technologies such as edge, cloud computing, V2X, and IoT into traffic monitoring and management [14]. V2X technology connects vehicle and road infrastructure, making adaptive real-time signal control at intersections possible.

On Board Unit (OBU) is the main application of vehicle side, each CV has an OBU, which mainly realizes the collection of vehicle status information. Road Side Unit (RSU) is the main application product on the road side, deployed at each intersection, mainly detecting the status information of itself or the surrounding environment, such as ETC and other non-contact sensing devices, and receiving and feeding back communication data through microwave. The controller mainly implements specifically adapted traffic control algorithms. For example, at a queuing lane, the OBU transmits CV position, speed, heading and acceleration information to the RSU in the form of Basic Safety Messages (BSMs), and consecutive BSMs represent the vehicle trajectory. At the same time, the RSU and the controller broadcast the signal timing plan to the traffic monitoring center for area-wide coordination [15].

In the ITS environment, the crossed vehicle states can be obtained in real time by the track information transmitted by the networked vehicles or by the cameras. The state volume should try to use real-time information from all vehicles as much as possible, and in general, using more information to describe the state can help the network to determine the value of the state more precisely. One type of state description is raster occupancy [16, 17]. The working principle is shown in Fig. 1:



**Fig. 1.** Environment perception of raster occupancy, for each lane approaching an intersection in the diagram, divide it into multiple cells. If a cell is occupied by one car, that cell will return 1, indicating that there is a car in that cell, otherwise it will return 0.

This state description method provides a large amount of information to the controller, almost similar to image data input. Therefore, this method is expected to be perfectly integrated with the image recognition vehicle detector.

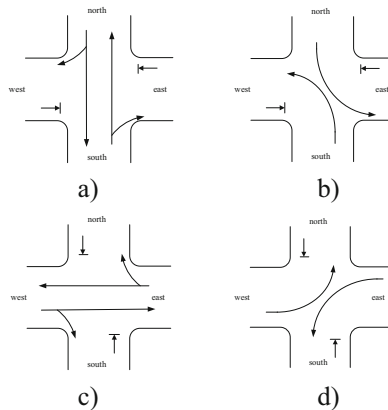
In an intelligent transportation system, it is assumed that an attacker is able to tamper with the status information obtained in the RSU. There are two main ways to do this attack, firstly, the attacker is able to access the internal network of V2X and the attacker tampers with the state input in the RSU. Secondly, the status can also falsify the CV trajectory at the end of the queue to create the illusion that a vehicle is parked at the end of the queue.

A state table attack is defined as a direct modification of the state table so that all lanes of an intersection are filled with networked cars, where the state of all lanes is tampered with, and the map of the occupied grid will change from 0 to 1. The state table attack is a direct attack that must invade the controller's interior, and this attack is more difficult to implement. There are few quantitative evaluations of information attacks, and the specific impact such an attack would have on the traffic control system is something that needs to be evaluated, such as how the attacker would attack to maximize overall traffic congestion.

### 3 RL Signal Controller for Intersection Suffering from Information Attack

#### 3.1 Reinforcement Learning Model Controller

**Intersection Scene Description.** In this paper, a typical urban road intersection controlled by traffic signals is considered.



**Fig. 2.** Cross-port phase setting. There are four traffic signal phases available, and after executing a certain phase, the corresponding priority time interval is assigned to a set of non-conflicting vehicle movements.

As shown in Fig. 2, there are four traffic signal phases available, and after executing a certain phase, the corresponding priority time interval is assigned to a set of non-conflicting vehicle movements. The lanes available for each of these phases are marked in green. This control process can be formulated as a reinforcement learning (RL) problem. The environment is a Markov process and reinforcement learning model learn by interacting with the environment. In this case, our environment is an isolated signal intersection.

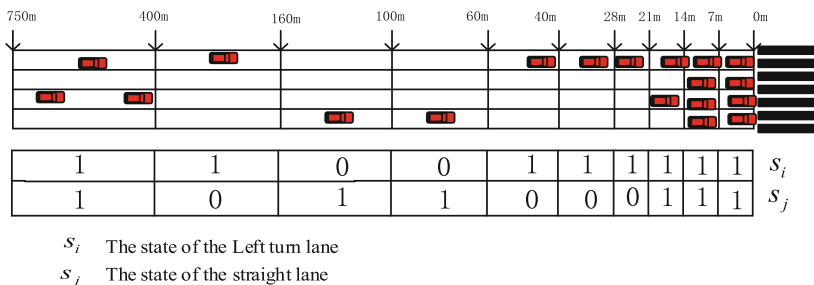
**Single Intersection Reinforcement Learning Model.** In RL, there are three important elements ( $S, A, R$ ) in the training process, where is the state space, ( $s \in S$ ),  $A$  is the action space, ( $a \in A$ ),  $R$  is the reward function. In the case of traffic signal control, we define these elements as follows:

**State:** The input data is a description of the current state. In general, using more information to describe the state can help the network to determine the value of the state more precisely. One of the state descriptions uses raster occupancy, which is described in Sect. 2.2.

For the traditional grid division, it is usually divided by equal distance, but the intersection is usually several hundred meters, and the equal distance division will lead to too large feature state set, which affects the calculation. This subsection makes some improvements to this state input, taking into account the arrival distribution of vehicles in the traffic flow, and divides the grid by distance. Specifically, the grid is divided in front of the frequently formed queue according to the length of the vehicle, and the length of the grid increases as the distance increases. The intervals are [0, 7], [7, 14, 21], [21, 28], [28, 40], [40, 60], [60, 100], [100, 160], [160, 400], [400, 750], which ensure that the focus is on the vehicle information in the queue, but also to ensure that we do not lose the information of the vehicles that are moving at high speed.

Similarly, for lanes that are in the same phase, we can combine them into the same state table since their traffic distribution is similar and they have the same travel route.

The detailed working principle is shown in Fig. 3. That is, whenever there is a vehicle in one lane, this grid is marked as 1, unless there is no vehicle in the straight lane of the same grid, then it is marked as 0.



**Fig. 3.** Intersection state awareness. That is, whenever there is a vehicle in one lane, this grid is marked as 1, unless there is no vehicle in the straight lane of the same grid, then it is marked as 0.

For each entrance lane, the state can be defined as  $s_i$ ,  $s_j$  is an array of 10 elements recording whether a vehicle occupies or not, and  $i$  represents its lane number. Then the set of states of each lane constitutes the whole set of states  $S$ .

$$S = \{s_1, \dots, s_i, \dots, s_8\} \quad (i = 1, 2, \dots, 8) \quad (1)$$

**Action:** The action here means selecting a signal phase. The possible signal phase configuration settings are shown in Fig. 3. The smart body picks up one of these four phases every second. The set of actions is as follows:

$$A = \{1, 2, 3, 4\} \quad (2)$$

Because the green light possesses a time interval limit, if the intelligence changes the current phase. Conversely, if the intelligent body keeps the current phase, it can choose the same signal phase action.

**Reward:** the reward is a guide that allows an intelligent body to perform an action in the right direction. In this case, we choose the difference between the cumulative vehicle wait times of two adjacent actions as the reward. The vehicle waiting time is defined as the time spent waiting in the vehicles in the import lane since the vehicles emerged from the environment.

$$W_t = \sum_{i=1}^{16} w_t^i \quad (3)$$

$W_t$  indicates the delay time for vehicles in all lanes, and  $w_t^i$  denotes the cumulative waiting time for all vehicles in the lane  $i$  at the time  $t$ .

Since the intelligence is engaged in a continuous decision process, for each state action, an immediate reward is generated, called  $R_t$ . Here we use the difference between the cumulative vehicle wait time  $W_{t-1}$  for the previous action  $a_t$ . And the cumulative vehicle wait time  $W_t$  for the current action  $a_t$  as  $R_t$ . The following is shown:

$$R_t = W_{t-1} - W_t \quad (4)$$

It means that  $R_t$  will encourage the intelligence to make the current action towards a smaller cumulative vehicle wait time than past actions, so as to ensure a larger reward and thus reduce the overall cumulative vehicle wait time.

**Q-Network Design.** Unlike model-based control methods that require full information about the environment, RL intelligences can learn optimal strategies for traffic signal control by interacting with the traffic environment. We use Q-networks in RL approach. It's a well-designed neural network is used as a Q-function approximator.

The goal of the Q-network is to train traffic signals to adapt to the phase and phase duration of intersections based on their real-time traffic patterns. This can be achieved by selecting an action in each training step ( $a_t \in A$ ) that maximizes the expectation of future cumulative rewards.

$$Q^*(s_t, a_t) = \max E[R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \dots | \pi] \quad (5)$$

where  $\gamma \in (0,1)$  is a discount factor that represents the trade-off between future and immediate payoffs. The current policy  $\pi$  is defined as the probability of taking action  $a_t$

in state  $s_t$ . According to dynamic programming theory, the optimal Q-function in Eq. (5) can be rewritten in the form of Bellman’s equation, as shown in the following equation:

$$Q^*(s_t, a_t) = E[R_t + \gamma \max_{a'} Q^*(s_{t+1}, a')] \tag{6}$$

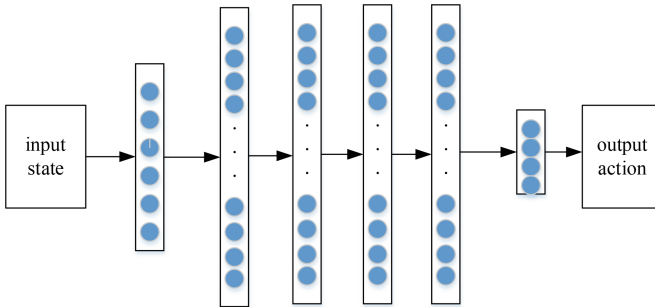
The traditional Q-learning algorithm solves the Bellman equation in an iterative manner, which relies on the concept of Q-table, which is a discrete table of Q-values. However, in our traffic signal control formulation, traditional Q-learning suffers from the curse of dimensionality because the state-behavior space becomes huge due to the existence of the definition of the state space[18].

Therefore, in this paper, we adopt the recent prominent idea of Q-network, which is to use neural network to approximate the Q-function:

$$Q(s, a : \theta) \approx Q^*(s, a) \tag{7}$$

where  $\theta$  denotes the parameter of the neural network.

Our Q-network for traffic signal control is shown in Fig. 4. At the beginning of each training step, state information is collected as the input layer of the neural network, and then feedforward is performed to estimate the Q-value at the output layer of the neural network.



**Fig. 4.** Q network structure diagram. It contains state inputs, six layers of neurons and action outputs.

To fit the function  $Q^*(s, a)$ , the deep neural network shown in Fig. 4 is used, and the network structure consists of six layers of neurons, with the first layer receiving the state  $S$  of the intersection, containing 80 neurons, and the middle four layers with 400 neurons. The last layer is the output layer and possesses four neurons that fit the action value function for each of the four phase actions. The middle neuron uses relu as the activation function, which has a good performance in fitting the nonlinear function. A linear function is used as the activation function for the final output. The maximum cumulative return is approximated effectively.

**Algorithm Process.** In the reinforcement learning model designed in this paper, first the model observes the state of the environment, and then the Q-network is used to fit the action value function and output the decided action, the environment generates a

new state at the same time, and so on and so forth. Until the best strategy is learned, which eventually makes the vehicle traffic flow at the traffic junction more efficient and achieves the purpose of reducing the delay.

In signal control at intersections, since the traffic flow state and phase settings are directly related, then there is a strong correlation between the samples  $(s, a, r, s')$  generated by the reinforcement learning model, which must be disrupted and redistributed in order to prevent overfitting of the deep neural network. To improve the stability of Q-network training, the mechanism of experience replay is used in our reinforcement learning. Specifically, experience replay is the storage of historical samples  $(s_t, a_t, r_t, s_{t+1})$  in a memory  $m$  that records historical samples. When the Q-network is trained, a certain number of historical samples are randomly selected as training data in each training process.

The exact process is shown in Algorithm 1:

---

Algorithm 1: Reinforcement Learning Signal Control Algorithm

---

1. Initialize Q network parameters and set parameter  $\theta$
  2. Initialization
  3. For epoch=1 to N do
    4. Initialize intersection state , action
    5. For 1 to T do
      6. Select actions  $a$  based on strategy  $\pi$
      7. Execute the action  $a$  , observe the reward  $r_t$  and the next state  $s_{t+1}$
      8. If (memory)  $m > M$ 
        9. Delete the oldest data in memory
      10. End if
      11. Store  $(s_t, a_t, r_t, s_{t+1})$  in  $M$
      12. Update  $\theta$
      13. From  $s_t$  to  $s_{t+1}$
    14. End for
  15. End for
- 

### 3.2 Signaling Attack Model Based on State Table

Assume that an attacker can directly manipulate the controller's state table input  $S = \{s_1, \dots, s_i, \dots, s_8\}$ . By directly modifying the state vector  $S$ , make the controller incorrectly believe that there are cars in the free lane.

Modify a phase lane state to  $s_a$ , so that each grid of this lane becomes 1:

$$s_a = \{1, 1, 1, 1, 1, 1, 1, 1, 1, 1\} \quad (8)$$

The complete state is generally collected by multiple RSUs, and we assume that the attacker has limited ability to attack and can only interfere with the state perception of certain lanes. To explore the impact of state attacks, a state table attack is designed to attack different lanes.



Attacking a single lane of traffic:

$$s_i = s_a \quad (9)$$

Attacking two lanes of traffic:

$$\begin{cases} s_i = s_a \\ s_j = s_a \end{cases} \quad (\{i, j\} \in \{1, 2, 3, 4, 5, 6, 7, 8\} \& i \neq j) \quad (10)$$

When  $i$  and  $j$  are in different relative positions, there are at least three different scenarios, i.e., (1) lane  $i$  and lane  $j$  are lanes in different phases in the same entrance direction, (2) lane  $i$  and lane  $j$  are lanes in different entrance directions and in different phases, and (3) lane  $i$  and lane  $j$  are lanes in the same phase.

The lane state is modified to measure the extent to which the controller is affected by the signal attack and to confirm whether this causes other congestion. Revealing the degree of impact of a information attacks can be an indication of the necessity of defense against potential information attacks.

## 4 Simulation Result

### 4.1 Simulation Settings

The simulation environment uses SUMO, a microscopic traffic simulation platform, to model intersections, and TraCI, a secondary development interface, using SUMO software and python language for secondary development, and TensorFlow framework for modeling and training of Q-networks. A joint SUMO-TensorFlow simulation platform was built to test and evaluate the signal control effects of the reinforcement learning model.

The intersection road network is built using SUMO's own road network editor, netedit, to create a two-way with 8-lane intersection, where the leftmost lane is a left-turn lane. The middle two lanes are straight lanes, the rightmost lane is a straight and right-turn lane, and the road length is set to 750 m.

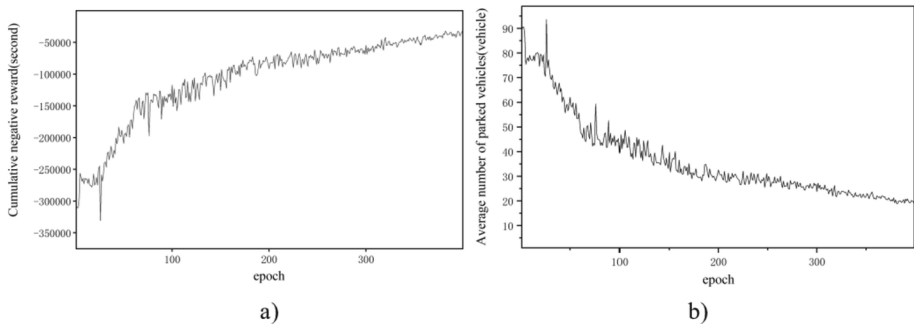
**Traffic Distribution:** For each simulation instance, the initial state is a traffic network with no vehicles, and then vehicles are randomly inserted at the destination and the corresponding path. Each simulation lasts for 1.5 h (5400 s). As we seek to propose a method that can be generalized to any case, the traffic distribution is Poisson to simulate the general case. That is, the traffic flow is 4500 veh/h. The left-turn and right-turn lanes are allocated 25% of the traffic flow.

**Signal Timing Scheme:** Since this scheme is a timing scheme without fixed phase sequence, the phase settings are the same as in Sect. 3.2, which is worth noting. After performing the control of the reinforcement learning model, the adjustment of the phase is completely determined by the reward, and it is up to the intelligence to decide the action to be taken for the next phase.

## 4.2 Reinforcement Learning Q-Network Training

First, we examine the simulation data, and the results show that our algorithm does learn good action strategies that are effective in reducing vehicle dwell time, thereby reducing vehicle delays and traffic congestion, and that our algorithm is stable in terms of control decisions. The controller does not oscillate between good and bad action strategies, or even favor bad action strategies.

Since we set the reward as the difference of vehicle waiting time between two adjacent actions, it is impossible to guarantee that the next action will always have less cumulative vehicle waiting time than the previous action since the traffic flow arrives randomly. In order to better evaluate the training process, only rewards with negative reward values are collected and summed in this paper, and the change of cumulative negative rewards reflects the whole process of training. As shown in Fig. 5a):



**Fig. 5.** a): Cumulative negative reward. b): Average number of parked vehicles.

As the number of times the model is trained increases, the cumulative negative reward value of the vehicles rapidly decreases the reward reaches convergence at around 370 turns. This indicates that the model does learn good action strategies from training. The same trend can be seen in Fig. 5b), where the average number of parked vehicles also remains stable at smaller values after 370 rounds, with a significant decrease in the number of vehicles in the queue. This indicates that the algorithm in this paper converges on a good action strategy and algorithmic stabilization mechanism, and that the empirical replay mechanism used by the model is effective.

## 4.3 Reinforcement Learning Control Model Results Analysis

The RL model does not need to follow a specific phase cycle, but runs acyclically. Each action is chosen to maximize the reward. In contrast to fixed-time control, which does not exhibit this goal-directed behavior and instead performs a relatively poor primitive logic. To evaluate the performance of our proposed reinforcement learning model, a fixed-time control strategy is used for comparison. The fixed-time control involves pre-calculating the signal phase duration based on the proportion of vehicle arrivals and fixing it for the duration of the run.

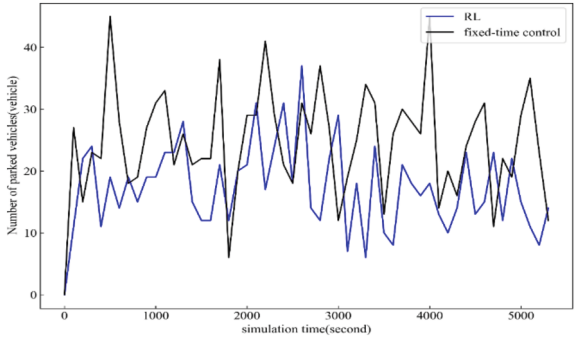


Fig. 6. Comparison of the average number of stops of the two controllers.

The difference in the number of vehicles in queue at each moment between the RL adaptive control strategy and the fixed-time control strategy can be seen in Fig. 6. The reinforcement learning control model significantly outperforms the fixed-time control strategy. Specifically, due to the effect of the adaptive control strategy of RL. The naked eye can intuitively see that most of the instantaneous queue lengths of the RL control model are lower than those of the fixed-time control queue lengths, with the maximum number of vehicles in the queue remaining stable at about 40 vehicles and the minimum at about 5 vehicles when in the RL control strategy. In comparison, the maximum queue of vehicles in the fixed-time control reaches 60, and the minimum is about 10 vehicles.

In order to better evaluate the overall control performance, the average queue length and total parking waiting time are used as evaluation metrics, as shown in Figs. 7a) and Figs. 7b), the number of queuing vehicles and total parking time are reduced by 23% and 22%, respectively, which proves the effectiveness of our reinforcement learning control model.

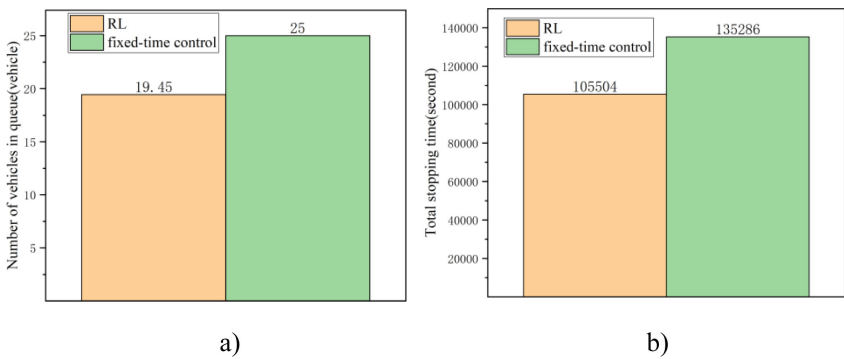


Fig. 7. Comparison of the control effect of the two controllers.

#### 4.4 Analysis of Signal Attack Results Based on State Table

On the basis of the established reinforcement learning model, the state is modified with corresponding attacks according to different scenarios. Similar to the previous section, we also used the average number of vehicles in queue and the total parking time as evaluation metrics. The specific results are shown in Table 1.

From the data in Table 1, it can be seen that, unlike what is imagined, the total average queue length improves by 7% if only one lane is attacked for state awareness, indicating that the attack (Type1) does not have a significant impact on the controller. However, when attacking the perception of two lanes and both lanes are in the same phase (Type2), the average queue length and total stopping time are significantly increased by 253%, which shows the significant interference of signal attack on the intersection controller. The results of attacking two lanes in different phases in the same entrance direction (Type3) and two lanes in different phases in different entrance directions (Type4) are improved by 60% and 19.6%, respectively. Both scenarios are in different phases, but there is also a big difference, which is because the same direction contains left-turn lanes, and the RL model assigns fewer actions to the left-turn lanes in order to release more vehicles to pass, when the left-turn lane falsifies the attack, making a large difference in the results.

**Table 1.** The degree of influence of the controller by the signal attack.

Attack Type	Total stopping time	Percentage increase in stopping time	The number of vehicles in queue	Percentage increase in vehicles in queue
No attack	105004	–	19.45	–
Type1	112237	7%	20.8	7%
Type2	371291	253%	68.75	253%
Type3	168843	60%	31.2672	60%
Type5	125639	19.6%	23.3	19.6%

The above results show that the RL model is robust and resistant to interference, and can still make suboptimal decisions even when a portion of state perception in the same phase is compromised by an attack. However, when the attacker attacks the overall perception of this phase, the impact on the system is particularly large, which confirms the great impact of network attacks on the signal control system. Fortunately, such direct attacks are easily identified by cyber security systems, but strengthening signal control system cyber security is also an urgent issue.

## 5 Conclusion

V2X, 5G and other communication technologies that connect vehicles, roads and infrastructure have undoubtedly brought new solutions to traffic signal control [19]. But at the same time, this extensive connectivity makes traffic signal systems very vulnerable to

information attacks [20]. The first step in how to protect against such attacks is to analyze how and to what extent such attacks affect signal controllers. In this paper, we evaluate the impact of the attack more objectively by building a model-free controller with reinforcement learning. We also argue that reinforcement learning not only has excellent control performance, but also has some robustness, which coincides with the literature that mentions reinforcement learning as the next generation of signal controllers.

For intersection signal controllers, the impact is huge if they are attacked. Conversely, if only some of the lane states in the same phase are attacked, the impact is limited and within an acceptable range, and a smart attacker will surely take advantage of this flaw [21, 22].

In the future it will be necessary to develop a series of defensive measures that will not only work on the level of network security to prevent malicious attackers from penetrating the internal network. It is also important to focus on the way attackers use personal vehicles to send fake tracks and establish reasonable screening methods. Finally, there is also a need for some degree of verification using data from other sensors, and a certain amount of signal redundancy is necessary.

## References

1. Wu, M., Xiong, N.N., Tan, L.: An intelligent adaptive algorithm for environment parameter estimation in smart cities. *IEEE Access* **6**, 23325–23337 (2018)
2. Xing, Y., Lv, C., Wang, H., et al.: Driver lane change intention inference for intelligent vehicles: framework, survey, and challenges. *IEEE Trans. Veh. Technol.* **68**(5), 4377–4390 (2019)
3. Florin, R., Olariu, S.: A survey of vehicular communications for traffic signal optimization. *Veh. Commun.* **2**(2), 70–79 (2015)
4. Li, W., Xu, H., Li, H., et al.: Complexity and algorithms for superposed data uploading problem in networks with smart devices. *IEEE Internet Things J.* **7**(7), 5882–5891 (2019)
5. Lee, J., Park, B.: Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment. *IEEE Trans. Intell. Transp. Syst.* **13**(1), 81–90 (2012)
6. Du, R., Qiu, G., Gao, K., et al.: Abnormal road surface recognition based on smartphone acceleration sensor. *Sensors* **20**(2), 451 (2020)
7. Lu, N., Cheng, N., Zhang, N., et al.: Connected vehicles: solutions and challenges. *IEEE Internet Things J.* **1**(4), 289–299 (2014)
8. Chen, Z., Luo, Z., Duan, X., et al.: Terminal handover in software-defined WLANs. *EURASIP J. Wirel. Commun. Netw.* **2020**(1), 1–13 (2020)
9. Zheng, G., Xiong, Y., Zang, X., et al.: Learning phase competition for traffic signal control. In: *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 1963–1972 (2019)
10. Zhao, T., Wang, P., Li, S.: Traffic signal control with deep reinforcement learning. In: *2019 International Conference on Intelligent Computing, Automation and Systems (ICICAS)*, pp. 763–767. *IEEE* (2019)
11. Chu, T., Wang, J., Codecà, L., et al.: Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE Trans. Intell. Transp. Syst.* **21**(3), 1086–1095 (2019)
12. Gao, J., Shen, Y., Liu, J., et al.: Adaptive traffic signal control: Deep reinforcement learning algorithm with experience replay and target network. *arXiv preprint arXiv:170502755* (2017)

13. Liang, X., Du, X., Wang, G., et al.: A deep reinforcement learning network for traffic light cycle control. *IEEE Trans. Veh. Technol.* **68**(2), 1243–1253 (2019)
14. Sumalee, A., Ho, H.W.: Smarter and more connected: future intelligent transportation system. *IATSS Res.* **42**(2), 67–71 (2018)
15. Alonso, B., Ibeas, Á., Musolino, G., et al.: Effects of traffic control regulation on network macroscopic fundamental diagram: a statistical analysis of real data. *Transp. Res. Part A: Policy Pract.* **126**, 136–151 (2019)
16. Marciandò, F.A., Musolino, G., Vitetta, A.: Signal setting optimization on urban road transport networks: the case of emergency evacuation. *Saf. Sci.* **72**, 209–220 (2015)
17. Filocamo, B., Ruiz, J.A., Sotelo, M.A.: Efficient management of road intersections for automated vehicles—the FRFP system applied to the various types of intersections and roundabouts. *Appl. Sci.* **10**(1), 316 (2019)
18. Lin, Y., Wang, P., Ma, M.: Intelligent transportation system (ITS): concept, challenge and opportunity. In: 2017 IEEE 3rd International Conference on Big Data Security on Cloud (bigdatasecurity), IEEE International Conference on High Performance and Smart Computing (hpsc), and IEEE International Conference on Intelligent Data and Security (IDS), pp. 167–172. IEEE (2017)
19. Liu, W., Zhou, Q., Li, W., et al.: Path optimization of carpool vehicles for commuting trips under major public health events. *J. Changsha Univ. Sci. Tech.* **1**, 112–120 (2023)
20. Zhang, Z.Y., Liu, Z., Jiang, L.: Follow-along model considering lateral spacing in vehicle networking environment. *J. Changsha Univ. Sci. Tech.* **3**, 62–68 (2021)
21. Balint, K., Tamas, T., Tamas, B.: Deep reinforcement learning based approach for traffic signal control. *Transp. Res. Procedia* **62**, 278–285 (2022)
22. Liu, W., Wang, Z., Liu, X., et al.: A survey of deep neural network architectures and their applications. *Neurocomputing* **234**, 11–26 (2017)