Complex & Intelligent Systems (2025) 11:217
https://doi.org/10.1007/s40747-025-01841-9

ORIGINAL ARTICLE

Traffic signal optimization control method based on attention mechanism updated weights double deep Q network

Huizhen Zhang[1] · Zhenwei Fang[1] · Youqing Chen[1] · Haotian Dai[1] · Qi Jiang[1] · Xinyan Zeng[1]

Received: 26 August 2024 / Accepted: 13 February 2025 / Published online: 19 March 2025

Abstract

As a critical guidance facility for vehicle convergence and diversion in urban traffic networks, the control effect of traffic signals directly affects traffic efficiency and road congestion level. As a mature deep reinforcement learning algorithm, the double deep Q network has shown a significant optimization effect in intelligent traffic signal control research. In this paper, for the feature extraction defects of deep double Q network and the problem of underestimating the evaluation value of actions, we propose an Attention Mechanism Updated Weights Double Deep Q Network (AMUW–DDQN) based on the attention mechanism for the optimal control of traffic signals. The AMUW–DDQN method enhances the perceptual ability of the network by introducing the attention mechanism of Squeeze And Excitation Networks (SENet) to make the neural network pay attention to important state components automatically, and based on the idea that accurate representation of potentially optimal action values is better than the balanced representation of all the action values, it is considered that underestimated actions have a certain probability of being the optimal action and the loss function is weighted to optimize the action values. Simulation experiments were also conducted using the traffic flow data of the intersection of Fengze Street–Tian'an South Road, Fengze District, Quanzhou City, Fujian Province, China. The experimental results show that the method proposed in this paper has the most significant final convergence effect for the same number of iterations, and has better performance in the evaluation indexes such as vehicle queue length and vehicle delay time.

Keywords Intelligent transportation · Traffic signal control · Double deep Q network · Reinforcement learning · Deep learning

Introduction

Traffic signals, as the primary method of intersection traffic control, play a crucial role in improving traffic flow efficiency through the implementation of a well-designed signal timing scheme. Periodic fixed timing is the mainstream method of traffic signal control, using the road's historical traffic data to adjust the traffic light time allocation of each phase. Such a control method is simple and effective, and can effectively reduce the queue length of vehicles at intersections where the characteristics of traffic flow are relatively fixed. However, in a large number of practical application cases, the fixed timing scheme shows obvious limitations and drawbacks for the actual nonlinear traffic flow scenarios, i.e., it is not possible to adaptively adjust the signal control scheme through real-time traffic flow information [1].

Adaptive traffic signal control methods have the capability to dynamically adjust the phase configuration scheme based on real-time traffic flow characteristics and optimization objectives. The continuous development of artificial intelligence technologies, such as machine learning, has advanced the research in the field of traffic control and made signal control with adaptive characteristics an important part of intelligent transportation. Deep reinforcement learning as an emerging research hotspot in the field of artificial intelligence, its excellent fitting ability of deep learning [2, 3] and decision-making ability in the traffic signal timing optimization method shows great potential for application, and the application of deep reinforcement learning to traffic signal control has gradually become a research hotspot nowadays [4–6].

Currently, reinforcement learning algorithms, particularly value-based Q-learning series algorithms, are the mainstream

✉ Huizhen Zhang
zhanghz@hqu.edu.cn

[1] College of Computer Science and Technology, Huaqiao University, Xiamen 361021, China

in traffic signal control. Since Li [7] and others first used deep neural network model to realize single-intersection signal optimization, domestic and foreign scholars proposed many new improvement methods for the research of value-based reinforcement learning algorithms applied to traffic signal scenarios and confirmed that reinforcement learning models are more effective in reducing traffic congestion in dynamic traffic scenarios are more effective in reducing traffic congestion [8, 9].

Swapno [10] and VM [11], as well as Xu [12], have utilized the DQN algorithm for adaptive traffic signal control, demonstrating through experimental comparisons in various scenarios that DQN effectively enhances traffic efficiency. Fan [13] addressed the problem of local and regional coordination in traffic signal control and constructed a model based on the DQN algorithm to improve the efficiency of the traffic in the designated area, and at the same time, reduce the time loss and waiting time of vehicles. Xu [14] combined Deep Q Network with Long Short-Term Memory (LSTM) to propose Deep Recurrent Q Network (DRQN) algorithm, and the experimental results show that fitting the Q-value function through LTSM network structure effectively reduces the average delay of vehicles. average delay. DDQN decouples the evaluation and selection of actions on the original network structure of DQN in a way that effectively mitigates the optimistic estimation of DQN and thus the evaluation bias problem. Experiments have proved that the network structure of DDQN is more suitable for signal control scenarios characterized by dynamic traffic changes [15–18].

In order to solve the traffic signal control problem in a large-scale adaptive traffic control environment, Zhang [19] combines the forgetting experience mechanism with the priority replay mechanism based on the DDQN algorithm to construct a forgetting priority-weighted double-depth Q learning algorithm, and the experimental results show that it has a better performance in terms of the vehicle speed, intersection delay, and intersection waiting queue length, etc. Bouktif [20] utilizes road information such as the number of vehicles and vehicle queuing time to simplify the definition of states and rewards in signal control and used the DDQN algorithm to demonstrate that his proposed method has obvious optimization effects on the number of vehicles in the queue and the total delay time, but the over-simplified description ignores the potential information in the state description. To address this problem, Ren [21] proposed a deep reinforcement learning algorithm based on the attention mechanism, which makes the neural network automatically pay attention to important state components to enhance the network's perceptual ability and to mine potentially important state features. Raeis [22] proposed a method based on the DDQN algorithm to realize the two concepts of fairness for the problem that fairness is ignored in traffic signal control, and through experiments three different vehicles to verify the effectiveness of the method for traffic signal control.Pálos [23] compared the effectiveness of DQN algorithms applied to traffic signal control optimization, and the experimental results found that the Dueling DQN [24] competing architectures have significant advantages in preventing unwanted phase-switching maneuvers, which effectively improves the stability of the intersection operating environment. Bhumeka [25] proposed SD-DQN based on the Dueling DQN (Sequential-Dueling Deep Q Network) model to alleviate traffic congestion in three-lane intersections. Sahu [26], Liang [27], and Gu [28] fused the features of DDQN and Dueling models to construct a signal timing based on D3QN (Double Dueling Deep Q Network) model, which utilizes multiple optimization elements to enhance the stability in traffic signal control environments. The experiments are based on state acquisition for discrete traffic changes and reward setting for cumulative vehicle delay changes, and the results show that D3QN has a faster convergence effect and optimization efficiency. Ni [29] utilizes the attention mechanism to emphasize the priority of the vehicles based on the D3QN algorithm, and the experimental results show the proposed algorithm has a better performance in terms of vehicle waiting time, queue length, and number of stopped vehicles. Dan Yang [30] combined D3QN and priority experience replay to effectively improve the performance of the model for traffic signal control. However, the improved algorithm based on D3QN has higher requirements for the training data and application environment of reinforcement learning intelligent agents, which leads to the difficulty of model training.

Traditional traffic signal optimization methods typically rely on manually set rules, which assume that traffic flow is static or exhibits minimal variability. As a result, these methods lack the ability to adapt to dynamic changes in the traffic system. With the increasing complexity and volume of urban traffic, traditional methods often fail to effectively address fluctuations in traffic flow and emergencies. This results in signal timings that cannot be adjusted in real time, thereby negatively impacting the efficiency and capacity of traffic flow. Although traffic signal optimization methods based on deep reinforcement learning (DRL) have shown significant improvements in single-intersection control and demonstrated superior performance in simulation environments, these approaches present several key challenges. First, DRL models typically require large amounts of training data and multiple interactions with the environment to converge, which can lead to high computational costs and long training times. This problem is exacerbated in complex traffic networks, where the training process is less efficient. Second, existing DRL methods often fail to fully capture the dynamic characteristics of traffic flow, as they tend to overlook potential features within the state information. This limitation reduces the model's ability to generalize effectively and undermines its stability in real-world traffic environments.

Third, in the context of traffic signal control, over-reliance on historical data or local features within the training set may cause the model to miss more effective global optimization strategies. This is especially problematic in high-dimensional and complex traffic systems, where the risk of converging to local optima is more pronounced.

In real-world traffic environments, the immediacy of road network control scenarios presents new challenges regarding the computational efficiency and effectiveness of the model. A key issue that needs to be addressed for the application of deep reinforcement learning (DRL) in intelligent signal control is how to construct an effective signal control model for experimental environments while integrating actual road characteristics for adaptive parameter design and algorithm optimization. To overcome the limitations of current traffic signal optimization methods, this paper focuses on improving the application performance of deep reinforcement learning in traffic signal control. Many existing studies employ value-based deep reinforcement learning approaches for traffic signal control, with Deep Q-Network (DQN) and Double Deep Q-Network (DDQN) being the most commonly used algorithms. However, these methods generally overlook the significance of state representations, often leading to local optima in complex traffic environments. Moreover, the reward function designs of these methods are typically simplistic, only accounting for basic metrics such as capacity and delay. To address these shortcomings, this paper proposes a single-intersection traffic signal DDQN model with a fixed phase order, enhanced by the attention mechanism. The model utilizes the difference in the dual-Q network's evaluation of actions as the basis for weight setting. When calculating the loss function, the weighted sum is computed according to the weights of the training samples to improve the model's accuracy in evaluating optimal actions. Additionally, discrete traffic coding is employed to capture information such as vehicle position and speed in each lane. Reward factors are defined based on the global average speed and the number of queuing vehicles, leading to the development of a traffic signal optimization control model, AMUW-DDQN, which integrates deep double Q-learning with updated weights. The main contribution of this paper:

1. To enhance the adaptability of the AMUW-DDQN model to diverse environments, the model is designed to incorporate the queuing vehicles from the original fixed time allocation as a factor in the reward function. This modification aims to improve the reasonableness of the reward function in action evaluation. Additionally, an attention mechanism is introduced to enable the neural network to automatically focus on the important state components, thereby enhancing the network's perceptual capabilities and uncovering potential key state features.
2. To improve the accuracy of optimal action value evaluation, the intelligent traffic signal control method based on AMUW-DDQN incorporates a differential weight setting derived from the double-Q network during inverse updates through the loss function. This adjustment effectively improves both the exploration capability and convergence of the model, thereby optimizing the intelligent traffic signal control strategy.
3. Simulation results demonstrate that, under identical training conditions, the intelligent traffic signal control method based on AMUW-DDQN proposed in this study exhibits a more pronounced optimization effect compared to existing methods. Notably, it shows significant improvements in evaluation metrics such as average delay time and queue length, thus validating the model's effectiveness and robustness in practical applications.

Problem definition

In the reinforcement learning-based traffic signal control environment, the signal lights on the road combined with intelligent control algorithms are defined as Agent, and the traffic flow that constitutes the lanes of the intersection is defined as the Environment. Through continual interactive learning between the intelligent agents and the external environment, these agents incrementally acquire the expertise to select actions yielding the highest expected returns within specific states. The interaction process between the intelligent agents and the environment in the realm of reinforcement learning algorithms is illustrated in Fig. 1.

The aforementioned framework highlights state, action, and reward as the fundamental components shaping the construction of a reinforcement learning algorithm. These key elements play a pivotal role in guiding the behavior of the agents and optimizing their strategic decisions [31]. Consequently, within the context of a reinforcement learning model applied to traffic signal control scenarios, precise identification of traffic flow data at road intersections and effective guidance for intelligent agents necessitate the establishment of tailored configurations for the core elements of reinforcement learning.

States setting

The state embodies the environmental information received by the reinforcement learning intelligence at each time step. In the scenario where traditional reinforcement learning algorithms are applied to traffic signal control, a tensor containing lane-level or roadway-level information is usually created as state information in the reinforcement learning traffic signal control method. The state information generally selects the queue length of the traffic flow in each direction or the overall

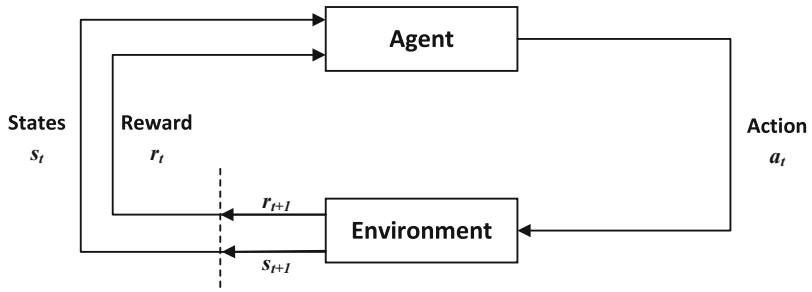

Fig. 1 Flowchart of the interaction between reinforcement learning agents and the environment

traffic flow as the state, and these scalar data can only roughly portray the state of the traffic environment in a single aspect, which makes the intelligent body unable to accurately perceive the actual situation in the outside world [32]. The main reason for the defect of dimensional catastrophe that exists in traditional reinforcement learning is that the scale of the state space of the environment grows exponentially with the number of external features of the environment. With the increasing ability of deep neural networks to model complex nonlinear systems, reinforcement learning algorithms incorporating deep learning utilize neural networks for higher dimensional abstraction and are able to construct evaluation networks based on convolutional neural networks and forward propagation networks, etc., which are applied to the traffic signal control task and then solve the value function.

In order to capture more accurate real-time traffic information, this paper adopts the widely recognized Discrete Traffic State Encoding (DTSE) method for intersection state acquisition [33]. Contrasting with traditional tabular representations of vehicle queue lengths and counts, DTSE provides a more precise portrayal of current vehicle positions and speeds [34]. The acquisition process of DTSE, as depicted in Fig. 2, entails dividing each approaching lane at the intersection into multiple squares positioned at fixed intervals from the front of the stop line. The length of these squares is set to accommodate only one vehicle, ensuring that each square can contain at most one car. By assigning binary values (0 or 1) to these squares based on vehicle presence, a position matrix representing vehicle locations is formed. Similarly, a speed matrix capturing real-time vehicle speeds is constructed. These matrices serve as inputs to the deep reinforcement learning model, enabling the network to generate Q-values for all actions in the action set. DTSE effectively captures traffic flow details from approaching lanes, reduces the computational burden of Q-value calculations through neural networks, and enables intelligent agents to uncover deeper internal features.

Action setting

The agent of the traffic signal obtains the traffic flow information of the current moment through DTSE and selects the best action from the action collection in accordance with a certain action selection strategy, and the environment changes or maintains the phase of the current signal under the effect of the executed action, to realize the control of the phase of the signal. In the model of intelligent traffic signal control, the way of phase switching is mainly divided into two ways: fixed phase sequence and flexible phase sequence. Although both methods are widely used in the actual road environment, in order to be more in line with realistic driving conditions and intersection scenarios, the switching of signal phases should have a fixed sequence to avoid irregular changes between different phases [35]. Overly flexible phase sequence control methods may result in longer waiting times for traffic in the idle direction in intersection scenarios where traffic flows are more uneven, ignoring the fairness characteristics of traditional signal control methods. In addition, most drivers are accustomed to driving scenarios with cyclic phase sequences when the application of fixed phase sequences of classical signals has matured, which may lead to dangerous driving behaviors when the changes in traffic signal sequences conflict with years of driving habits.

In the signaling control of this paper, a cyclic transformation of the phase order of the original signals fixed in the applied intersection scenario is used. The set of actions of the signaling intelligence is defined as A={0,1}. At any moment t of interaction with the external environment, if $a_t = 0$, the signal light keeps the current phase unchanged; if $a_t = 1$, the signal light switches to the next phase in the phase sequence, and the duration of each action is the time of one simulation *step_time*. In order to ensure the fairness and safety of the allocation of access rights to each phase, the shortest phase duration is defined as $green\_time_{\min}/s$, and the longest phase duration is $green\_time_{\max}/s$. At the same time, it is necessary to set up a certain length of yellow light time in the gap between the green light switching to the red light, to ensure that the stranded vehicles in the transition phase are emptied [36]. The duration of the yellow light does not exceed the time of the simulation step, and the remaining green light time of the signal switching is counted as the duration of the phase. Based on the above sequence of phases and set of actions, the intelligent body coordinates the scheduling of traffic signals.

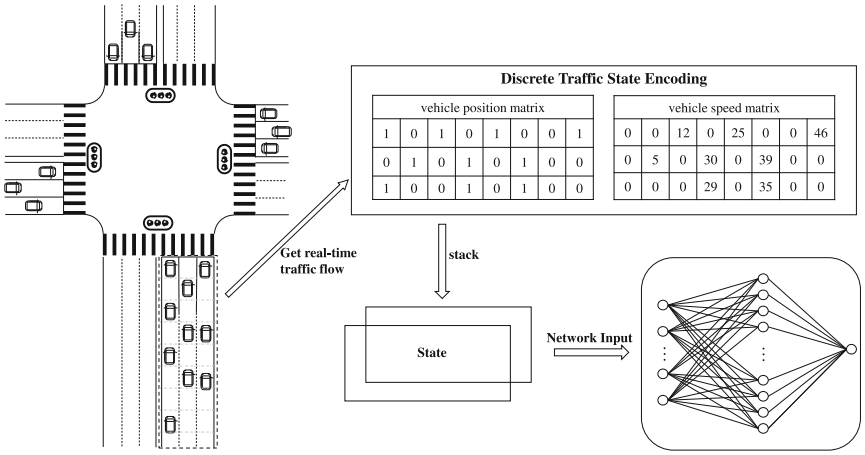

Fig. 2 Discrete traffic coding schematic

Reward setting

The reward function in reinforcement learning is crucial for determining the value of an action based on changes in the external environment state following its execution by the intelligent agent. In the realm of intelligent traffic signal control, the ability of the signal intelligence to learn optimal action strategies for enhancing vehicular traffic efficiency is intricately linked to the appropriateness of the reward function [37]. Consequently, various methods for setting reward values have been proposed for traffic signal control scenarios, such as those based on alterations in vehicle queue length, intersection throughput, and vehicle waiting time. Early studies in reinforcement learning often employed a single reward factor to gauge the degree of optimization achieved by executed actions. However, a singular objective may not guarantee that the intelligence adapts traffic signals to diverse requirements [38]. The efficacy of the reward function setting hinges on the primary objective of traffic control [39]. An effective signal control strategy should encompass a holistic view of intersection traffic conditions and manage delay costs within reasonable bounds based on overall traffic circumstances. In the research context of this paper, the reward function following the execution of an action by the intelligent agent is defined as shown in (1):

$$
\begin{aligned}
&reward_{total} = k_v \times reward_v + k_q \times reward_q \\
&s.t. \begin{cases} reward_v = v_{current} - v_{pre} \\ reward_q = vehicle_{base} - Queuing_{veh} \\ vehicle_{base} = ratio \times Queuing\_veh_{fixed} \end{cases}
\end{aligned} \tag{1}
$$

The definition of the reward function is divided into two parts: the first part $reward_v$ is the global vehicle speed difference between the time steps before and after the interaction environment, $v_{current}$ is the global average vehicle speed at the current moment, and v_{pre} is the global average vehicle speed at the previous time step, which utilizes the global speed difference between the adjacent time steps to reflect the smoothness of the vehicle traveling after taking action; the second part $reward_q$ is calculated through the average queuing number of vehicles under the original intersection fixed timing scheme $Queuing\_veh_{\text{fixed}}$ has a $ratio \in [0, 1]$, which is obtained by the difference with the number of real-time queuing vehicles on the road under the signal intelligent control scenario $Queuing\_veh$, and the average queuing number of vehicles under the original intersection fixed timing scheme is obtained by the simulation experiment of $Queuing\_veh_{\text{fixed}}$. number of vehicles, calculated as shown in (2):

$$
Queuing\_veh_{fixed} = \frac{1}{n_{step}} \sum_{i=1}^{n_{stsp}} queueVehicles_i \tag{2}
$$

where n_{step} denotes the number of times that the queuing vehicle information is acquired during the training process of a round of simulation, and $queueVehicles_i$ denotes the number of queuing vehicles globally at the ith time step. The intelligent control method is utilized to judge the action in comparison with the number of vehicles in the queue with fixed timing, and if the queued vehicles after the action taken by the intelligent control algorithm exceed a certain percentage of the queued vehicles under the fixed timing scheme, the $reward_q$ will generate a penalty term in order to avoid the iterative process from getting into trouble.

Based on the traffic elements of vehicle speed and the number of vehicles in the queue, the baseline of the reward function is set according to the traffic flow state in different traffic environments, and different weights k_v and k_q are assigned to it. The reward setting method based on AMUW-DDQN model is constructed by the above parameter setting method.

Attention mechanism based update weight deep double Q network model

SENet attention mechanism

The convolutional neural network relies on the convolution operator as its fundamental building block, enabling the network to capture informative features by integrating spatial and channel information within the local receptive field of each layer [40]. To effectively extract state information from the intersection road environment, the state data is initially processed using the SENet attention mechanism. This mechanism evaluates the significance of each feature channel through learning and perception processes, amplifying crucial state features based on their respective importance levels. The SENet attention mechanism comprises three key modules: squeeze, excitation, and scaling. The flow of operations within the SENet mechanism is illustrated in Fig. 3, demonstrating how the network assesses feature channel importance, adjusts feature representations, and scales the feature responses accordingly. By incorporating the SENet attention mechanism into the neural network architecture, the model can focus on and enhance essential state features, leading to improved information extraction and more effective decision-making processes.

Firstly, the squeeze module performs a global average pooling operation on the input feature map. This operation involves averaging the feature values across all spatial locations within each channel, effectively condensing the 2D features of the neural network into a single value per channel. The global feature representation for each channel is obtained through the squeeze operation, and the calculation for the global average feature value on channel C can be expressed as follows in equation (3):

$$z_c = F_{sq}(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i, j) \tag{3}$$

Following the global feature extraction in the squeeze module, the excitation component employs two fully connected layers to determine the importance weights for each channel. This process typically involves dimensionality reduction followed by dimensionality enhancement operations. The learned weights from these operations reflect the relative significance of each channel. The excitation operation incorporates the ReLU activation function and sigmoid activation function, where these functions play a role in shaping the feature values. The feature values post the excitation operation can be represented as follows in Eq. (4):

$$s = \mathbf{F}_{ex}(\mathbf{x}, \mathbf{W}) = \sigma\big(g(\mathbf{z}, \mathbf{W})\big) = \sigma\big(\mathbf{W}_2 \delta(\mathbf{W}_1 \mathbf{z})\big) \tag{4}$$

Finally, the SCALE section of the SENet attention mechanism performs a scaling operation on the features. This operation involves multiplying the learned weights with the original features, effectively assigning higher weights to more important features. This nonlinear transformation helps enhance the representation of the model and makes crucial features more prominent in the overall feature representation. By combining the squeeze, excitation, and scale components, the SENet attention mechanism enhances the neural network's capability to perceive important states. This improvement in feature representation leads to better performance of the model. The network structure of the SENet attention mechanism can be visualized in Fig. 4, demonstrating the flow and interactions of these three components within the architecture.

Update weight deep double Q network model

The Temporal Difference (TD) algorithm leverages the value function of the state at the next time step to approximate the value function of the current state, facilitating policy learning for incomplete trajectories. In the case of Deep Q-Networks (DQN), the algorithm utilizes the value function of the subsequent state or the action value function when updating the target value. To expedite the convergence towards an optimal solution, DQN employs a target network to determine the optimal action value for updating the model. The target value in the DQN algorithm is represented by (5):

$$y = \begin{cases} R & is_{end} \\ R + \gamma max_{a'} \widehat{Q}(s', a'; \theta^{-}) & is_{not_{end}} \end{cases} \tag{5}$$

where θ^{-} is a parameter of the target network, s' and a' are the actions corresponding to the state of the next time step and the maximum evaluation value of the target network, and γ is the attenuation factor. Since the process of selecting an action based on the state of the next time step in the DQN uses the same network model parameters as predicting the Q-value of the next time step, it makes the evaluating network's evaluation value for the state higher than the true level, and the error for the evaluation increases as the number of training samples increases thus [41]. The overestimation of the evaluation value of different actions by the DQN is not balanced, and the more frequently selected actions are overestimated, which ultimately leads to the probability of a suboptimal action being selected over the optimal action. In traffic signal control scenarios, if the phase of the current state is incorrectly judged to perform a switching action resulting in the execution of an unreasonable phase signal, it will increase the overall delay time of the vehicle and may even increase the risk of traffic accidents. In order to improve the accuracy of DQN for state evaluation, the DDQN algorithm decouples the next time step action selection and action state

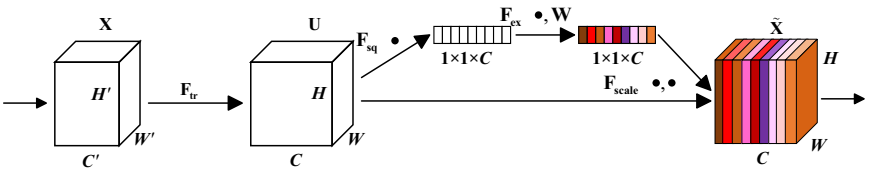

Fig. 3 SENet attention mechanism process

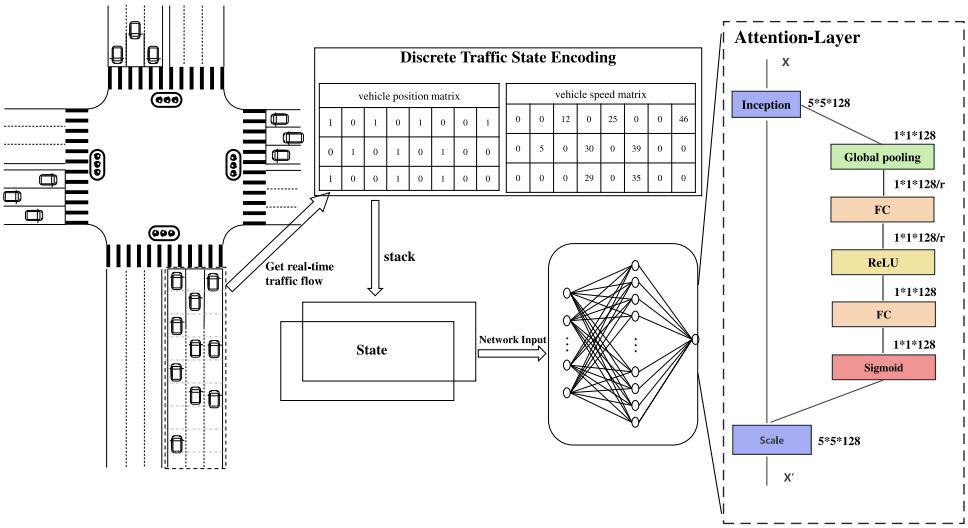

Fig. 4 SENet attention mechanisms network structure

estimation and uses the two networks of the original DQN architecture to perform the action selection and evaluation separately, which reduces the direct correlation between the calculation of the target value and the selection of the action. The formula for the target value of the DDQN algorithm is shown in (6):

$$y = \begin{cases} R & is_{end} \\ R + \gamma \widehat{Q}(s', \arg\max_{a'}(Q(s', a', \theta)); \theta') & is_{not_{end}} \end{cases} \quad (6)$$

where θ is a parameter of the evaluation network. In the algorithmic model of DDQN, the action value of the action set in the current state is first calculated from the evaluation network, and the one that has the largest action value among the above action values is selected, and then the action is used to calculate the target Q value in the target network. Although the improvement of DDQN can effectively control the risk of overestimation of action values in DQN, underestimation of action estimates may arise due to the lag in the synchronization of the two isomorphic networks of the model [42]. Even though the likelihood of underestimation or overestimation of the two estimators in the same action is low, it will affect to some extent the accuracy of the model's state estimation when the model converges. In this study, the discrepancy between the two estimators mentioned earlier is leveraged as a key point for enhancing the model.

Deep reinforcement learning requires a large number of empirical samples to support the training process of the network model, and the traffic signal control environment, as a scenario with a large sampling cost of empirical samples, has higher requirements for the training efficiency of the samples. To more accurately represent the evaluation value of the optimal action, this study is based on the idea that accurate representation of the potential optimal action value is better than balanced representation of all the action values, the proposed update weights based on the attention mechanism of the deep double Q network model AMUW-DDQN introduces the attention mechanism to enhance the neural network's ability to sense the important state, and will be weighted for each action. In the backward updating of the network parameters through the loss function, higher weights are assigned to the possible optimal actions, and lower weights are assigned to the other possible sub-optimal actions. The model sets the action with objective value y greater than $Q(s, a; \theta)$ as the underestimated action, i.e., it is likely to be the optimal action in the current state, and the model will tend to calculate the optimal action more accurately when optimizing with this weighted objective function. When the weight setting is constant at 1, the model will degrade to a normal DDQN model. So the motivation for introducing the weights w(s,a) is that they can be used to adjust the importance of different state-action pairs to better adapt to the traffic environment so that the model can mitigate the problems arising from overestimation, thus improving the training efficiency and performance performance. The weight setting formula of AMUW-DDQN

is shown in (7):

$$w(s,a) = \begin{cases} 1 & Q(s,a;\theta) < y \\ \delta & otherwise \end{cases}$$
$$s.t. \begin{cases} \delta = \frac{y}{Q(s,a;\theta)} \\ \delta < 1 \\ \gamma = R + \gamma \widehat{Q}\left(s', \arg max_{a'}\left(Q(s',a',\theta)\right); \theta\text{-}\right) \end{cases} \quad (7)$$

The setting of weights is calculated by evaluating the evaluation value $Q(s,a;\theta)$ of the network for the current state-action pair and the size of the target value. If the evaluation value is smaller than the target value, the action a has a certain probability to be the optimal action in the current state s, so a higher weight value of 1 is set; if the evaluation value is larger than the target value, it means that the model has tended to be accurate in evaluating the action in the state, and the weight is set to be the ratio of the target value to the evaluation value. At the same time, the minimum value of weight δ min is set to ensure the diversity and balance of the model training samples. The loss function of the model is optimized by the above weight settings, and the formula of the model loss function is shown in (8):

$$Loss_{AMUW-DDQN} = \sum_{i=1}^{b} w(s,a)(Q(s,a;\theta) - y)^2 \quad (8)$$

After weighting the empirical samples, the loss function of the AMUW-DDQN model places more emphasis on accurately estimating the optimal actions, leading to improved training efficiency and performance in the traffic signal control scenario. This is achieved by prioritizing the inverse optimization of the network parameters for the potential optimal actions, rather than maintaining a balanced representation of all action values. Compared to the original DDQN model, the AMUW–DDQN model no longer trains the evaluation value of the current state for all actions in a balanced way. Instead, it focuses on estimating the optimal actions evaluated in the current state, which effectively alleviates the problem of underestimation of the action value by DDQN. The method of solving the objective value in the AMUW–DDQN framework is consistent with that of the original DDQN. The model architecture of the AMUW-DDQN algorithm is shown in Fig. 5, which incorporates the attention mechanism and weighted objective function into the standard DDQN model.

In particular, the network structure of the evaluation and target networks of AMUW-DDQN consists of three convolutional layers and two fully connected layers. After the output feature map of the convolutional layers, the SENet module is applied to introduce an attention mechanism, which strengthens the model's focus on important features. The output features from SENet are then connected to the inputs prior to the fully connected layers to pass the attention-weighted features to the fully connected layers for further processing and action decision generation. The first convolutional layer consists of 32 filters with a 4×4 kernel size and a stride of 4. The second layer has 64 filters with a 2×2 kernel size and a stride of 2. The third layer contains 128 filters with a 2×2 kernel size and a stride of 2. After the convolutional layers' output feature map, the SENet module is introduced, which adaptively assigns different weights to each channel to enhance the model's focus on critical features. The fourth hidden layer is a fully connected layer consisting of 512 neurons, followed by the output layer, where the number of neurons corresponds to the size of the action space, and this layer is responsible for generating the final action decisions. The activation function for the first four layers is the rectified linear unit (ReLU), and the number of neurons in the output layer is the same as the number defined by the action space.

To ensure the precision of AMUW-DDQN during the dual network update process using weighted concepts, the algorithm initially adopts the conventional DDQN updating method. Upon reaching a specified global training step size η, the algorithm transitions to the weight-based update strategy. These configurations serve to prevent AMUW-DDQN from executing focused parameter updates during model instability, thereby enhancing algorithmic stability.

Experiments

Experimental data

The simulation experiment uses the intersection of Fengze Street-Tianan South Road in Fengze District, Quanzhou City, Fujian Province as an example. This intersection has four lane directions: southeast, west, north, and west. The specific intersection configuration is shown in Fig. 6.

The period of the original signaling timing scheme is 141 s. The number of phases is 4. The first phase duration is 32/s, the second phase duration is 32/s, the third phase duration is 32/s, and the fourth phase duration is 25/s. The phases are shown in Fig. 7.

The experiment selected off-peak traffic data for the morning time period of 09:00-10:00 and peak traffic data for the evening time period of 17:30-18:30 on August 23, 2019. The data was obtained by analyzing intersection videos and generating traffic flow text data. The intersection traffic video data was provided by the Quanzhou Municipal Bureau of Transportation in MP4 file format. The video data was manually analyzed to count the number of cars and convert it into traffic flow text data. In the proposed solution for the intersection timing scheme, the historical traffic data corresponding to the time period of the road is used as a model input, while the simulation time traffic data corresponding

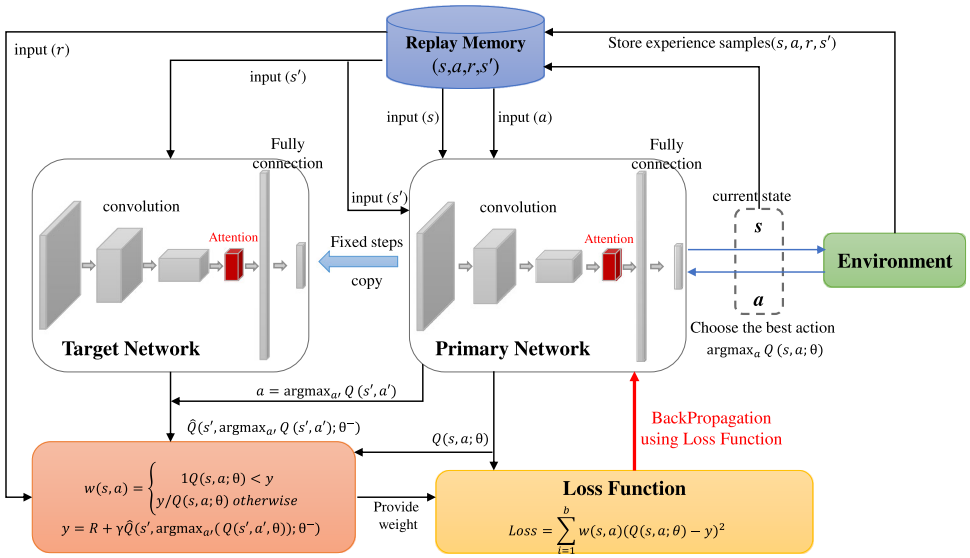

Fig. 5 AMUW-DDQN algorithm modeling

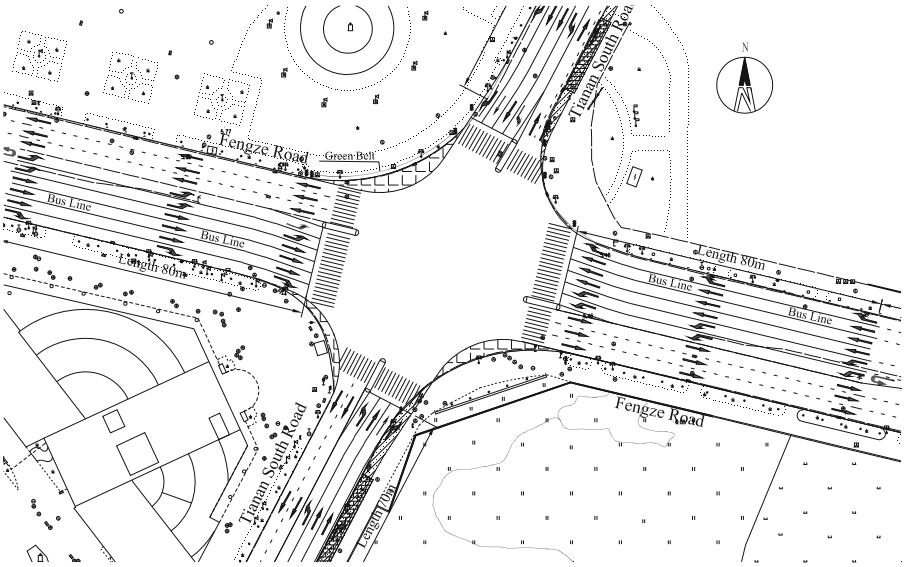

Fig. 6 Schematic diagram of the Fengze-Tianan intersection

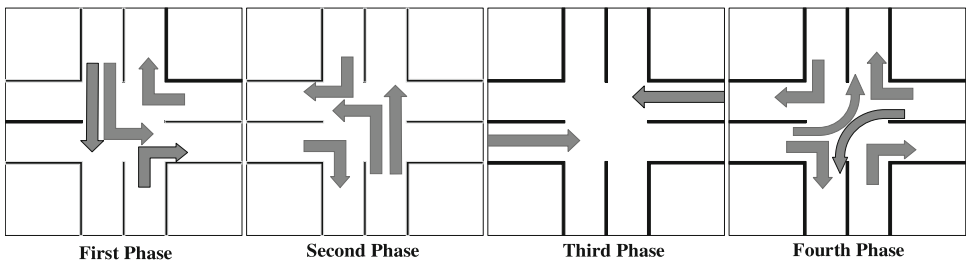

Fig. 7 Traffic signal movement and phase sequence

to the signal timing scheme is used as a model output. The traffic data during the simulation time period is presented in Table 1.

The traffic signal control algorithm based on AMUW-DDQN focuses on monitoring the road section located 200 ms in front of the parking line across the four approaching lanes at the simulated intersection. It sets the discrete traffic coding unit's division length as 5 ms and conducts the simulation process at a rate of 30 min per iteration. To better reflect actual traffic operations, a 3-second yellow light transition phase is incorporated between each phase switch to clear vehicles that have not yet exited the intersection. In the signal control scenario, the AMUW-DDQN model acquires real-time traffic flow information from the roadway approach via DTSE. It defines the decision to switch to the next phase as the set of intelligent actions and constructs a real-time intel-

Table 1 Traffic Data Information Sheet for Fengze-Tianan Intersection

Traffic flow data/vehicle		From South to North	From East to West	From North to South	From West to East
Flat-peak	Left Turn	281	191	175	154
	Straight Ahead	348	670	346	532
	Right Turn	348	670	346	532
Peak	Left Turn	349	295	245	189
	Straight Ahead	644	911	699	1025
	Right Turn	216	294	195	261

ligent signal control scheme based on changes in the total queued vehicles and the global average speed, which serve as the rewards following the execution of these actions.

In this study, the flow data is configured in accordance with the input specifications of the Vissim traffic simulation software, and a simulation environment running at a rate of 30 min per iteration is established based on the traffic flow characteristics outlined in Table 1. Additionally, various types of sensors are placed along each approach lane in the road network profile to capture real-time vehicle information, which is used as the data source for the comparison of evaluation indexes.

Evaluation indicators

In the AMUW-DDQN-based intelligent signal control experiment, the key metrics utilized to assess the correctness and precision of the model primarily consist of the average vehicle delay and the average queue length. The definitions of the evaluation metrics are detailed below:

- Average Vehicle Delay: the average of the difference between the actual travel time of vehicles on all roads within the simulation and the travel time in an ideal environment where there is no influence of other vehicles and there is no signalized interception, as shown in (9).

$$Delay\_average = \frac{1}{n_{\text{veh}}} \sum_{i=1}^{n_{\text{veh}}} Delay\_veh_i \tag{9}$$

  where n_{veh} denotes the number of all cars passing through the intersection under the complete simulation process and $Delay\_veh_i$ denotes the delay time of the ith car at the intersection.
- Average queue length: the mean value of the length constituted by the queue of vehicles meeting the queuing speed in each approach lane in the intersection under the complete simulation process, as shown in (10).

$$Queue\_average = \frac{1}{n_{lane}} \sum_{i=1}^{n_{lans}} Queue\_lane_i \tag{10}$$

  where n_{lane} denotes the number of all incoming lanes in the simulation environment and $Queue\_lane_i$ denotes the average queue length of the ith lane throughout the simulation process.

Experimental parameters and environment settings

Before the simulation training of a single intersection, the initial parameters of AMUW-DDQN need to be set, and the settings of the parameters after comprehensive consideration are shown in Table 2. The initial phase of the simulation iteration will be preceded by the filling operation of the empirical playback pool, this paper sets the amount of data to be filled in the empirical playback pool at the beginning of the training of the model to be 1/10 of the total capacity, and the evaluation data produced by this filling process will not be included in the final results of the experiment in the statistics.

In this paper, the experimental comparison in this study was conducted in a hardware environment equipped with an Intel Core i7-11700 CPU and an NVIDIA GeForce RTX 3060 12GB GPU, which supports CUDA parallel computing. Traffic simulation and modeling were performed using VISSIM 4.30, a widely used traffic simulation tool known for its efficient simulation capabilities and flexible control interfaces. VISSIM generates detailed and accurate simulation results, visualizing the operations of traffic systems while accounting for factors such as vehicle interactions, acceleration, deceleration, and car-following distances. In addition, VISSIM provides a COM interface for traffic signal control and the retrieval of various traffic data through different programming languages. The optimization of traffic signals based on the AMUW-DDQN approach was implemented using the TensorFlow-gpu 2.4.0 deep learning framework, with simulations conducted on a single intersection for optimization control.

Analysis of experimental results

To demonstrate the impact of AMUW-DDQN on the signal control effect, this paper performs comparative experiments involving mainstream DQN series models under identi-

Table 2 Experimental parameterization of AMUW-DDQN at a single intersection

Parameter	Value	Parameter	Value
Replay Buffer Size	10240	State Matrix Size	[89,86]
Learning Rate	0.001	Action Space Size	2
Batch Size	128	Number of Iterations	200
Discount factor	0.99	Simulation period	1800 s
Frequency of target network updates	100	Metameric length	5 m
$step\_time$	5/s	$ratio$	2/3
$\delta_{\min}$	0.85	Number of Iterations	$0.9 \rightarrow 0.01$
k_v k_q	0.2, 0.1	η	1000

cal conditions. The impact of model enhancements can be assessed through metrics such as average vehicle delay and average queue length within the traffic simulation environment. All comparative simulation experiments consist of 200 training rounds, and the experimental outcomes are illustrated in Figs. 8 and 9.

The experiments in this paper record the average vehicle delay and average vehicle queue length for different models at each iteration round under the same traffic environment. When comparing the computational complexity of the models, the AMUW-DDQN model proposed in this study incorporates an attention mechanism module, which mainly adds global average pooling and computes attention weights. Additionally, the update weights are calculated using the output of the two networks. An analysis of the data distribution reveals that the Dueling-DQN and D3QN models exhibit oscillatory behavior in the later stages of training, which prevents the average delay and queue length from being maintained at low levels during the final stages of simulation. The poor convergence of these models results in suboptimal signal control. In contrast, the AMUW-DDQN model demonstrates a more pronounced convergence trend compared to the other algorithms, with a more stable optimization effect in the later stages of training. The average delay and queue length values are kept within a smaller range, indicating that although the loss function increases computational complexity due to the introduction of the SENet attention module and the computation of attention weights, the AMUW-DDQN model achieves stable convergence starting at around the 60th training round. This suggests that the AMUW-DDQN model is more effective in terms of convergence compared to the other models. The experimental results of the latter 20 rounds under the flat and peak datasets are analyzed as shown in Tables 3 and 4.

Compared to the AMUW-DDQN model proposed in this paper, DQN, DDQN, Dueling-DQN, and D3QN all show obvious limitations. AMUW-DDQN ultimately has the most obvious convergence effect for the same number of iterations, with average delay and queue length evaluation metrics improved by 51.24% and 43.58%, respectively, compared to the fixed allotment, and with the next best DDQN model by 1.8% and 2.31%, respectively. In the experiments under the peak dataset, the average delay and queue length evaluation metrics are improved by 55.64% and 51.59%, respectively, compared to the fixed allotment, and by 3.10% and 2.99%, respectively, compared to the suboptimal Dueling-DQN model. Meanwhile, it reflects excellent optimization effect under the worst case control, ensuring that when facing intricate traffic conditions, AMUW-DDQN can guarantee the optimization of passage efficiency while reducing the probability of poor control effect of the intelligent signal control algorithm.

Conclusion

In this paper, we propose a deep reinforcement learning model AMUW-DDQN based on the attention mechanism for updating the weights of deep dual-Q networks, which not only introduces the attention mechanism in the target network and evaluation network to improve the neural network's ability to perceive important potential state features but also improves the model's ability to optimize the optimal action values by introducing the setting of the disparity weights in the inverse updating of the loss function to evaluation accuracy and reduce the probability of underestimating the suboptimal action. In addition, based on the importance of the three key elements of reinforcement learning, the reward function with adaptability to different environments is designed by combining the vehicle state information under fixed allocation time. The above settings realize the optimal control effect of AMUW-DDQN compared with other mainstream algorithms of DQN series.

The limitation of this paper is that only single-intersection signal timing optimization is considered. However, in real-world traffic control scenarios, multiple intersections are interconnected. Facing a large-scale urban traffic network with strong correlations of traffic flows between adjacent intersections, the AMUW-DDQN traffic signal control method based on AMUW proposed in this paper is not capa-

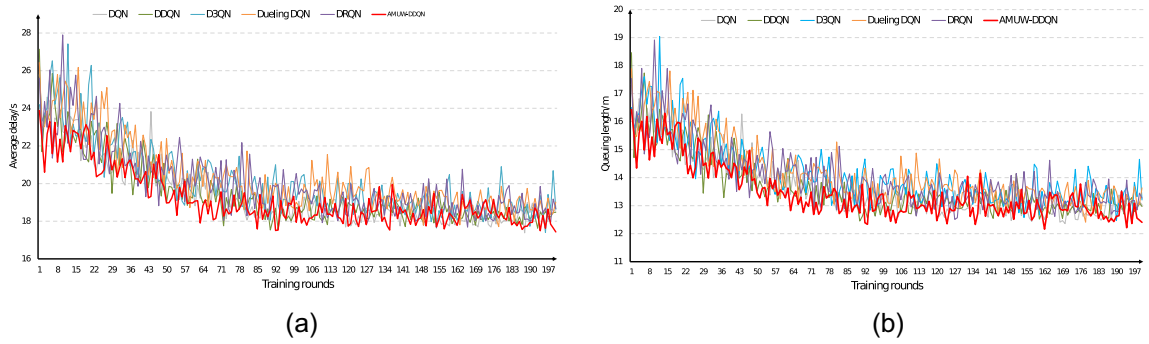

Fig. 8 Comparison plot of experimental results under the flat-peak data set. **a** Comparison of average vehicle delays under the flat-peak dataset. **b** Case Comparison of average queue lengths under the flat-peak dataset

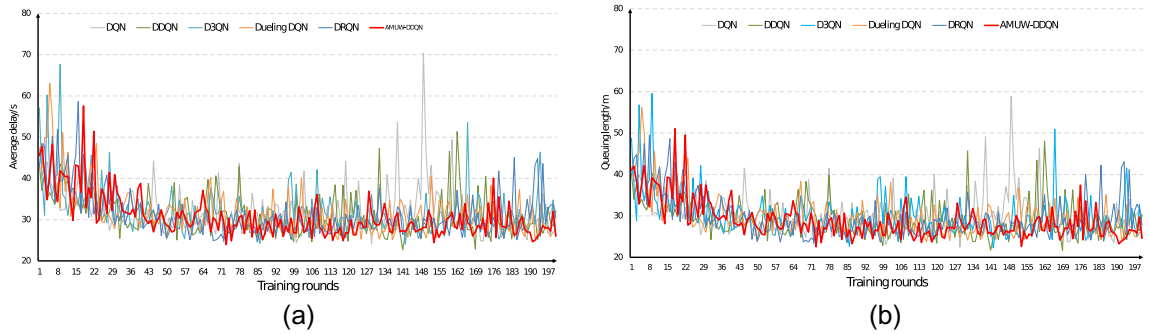

Fig. 9 Comparison plot of experimental results under the peak data set. **a** Comparison of average vehicle delays under the peak dataset. **b** Comparison of average queue lengths under the peak dataset

Table 3 Analyzing the experimental results based on the flat peak traffic dataset

Indicators		DQN	DDQN	Dueling-DQN	D3QN	DRQN	AMUW-DDQN	Fixed
Delay	Average value	18.33/*s*	18.29/*s*	18.83/*s*	18.71/*s*	18.68/*s*	**17.96/*s***	36.83/*s*
	Optimal value	**17.38/*s***	17.63/*s*	18.03/*s*	17.40/*s*	17.94/*s*	17.43/*s*	
	Least value	18.99/*s*	18.82/*s*	19.75/*s*	20.68/*s*	19.89/*s*	**18.72/*s***	
Queue	Average value	12.90/*m*	12.98/*m*	13.29/*m*	13.34/*m*	13.21/*m*	**12.68/*m***	22.44/*m*
	Optimal value	12.47/*m*	12.98/*m*	12.71/*m*	12.42/*m*	12.69/*m*	**12.21/*m***	
	Least value	**13.39/*m***	13.60/*m*	13.71/*m*	14.65/*m*	13.95/*m*	13.50/*m*	

Table 4 Analyzing the experimental results based on the peak traffic dataset

Indicators		DQN	DDQN	Dueling-DQN	D3QN	DRQN	AMUW-DDQN	Fixed
Delay	Average value	30.07/*s*	29.15/*s*	28.83/*s*	30.89/*s*	32.99/*s*	**27.93/*s***	62.90/*s*
	Optimal value	25.08/*s*	25.23/*s*	25.59/*s*	26.32/*s*	25.13/*s*	**24.66/*s***	
	Least value	36.42/*s*	33.08/*s*	37.30/*s*	46.38/*s*	45.07/*s*	**34.47/*s***	
Queue	Average value	28.40/*m*	27.54/*m*	27.26/*m*	28.94/*m*	31.21/*m*	**26.44/*m***	54.63/*m*
	Optimal value	23.57/*m*	24.12/*m*	23.98/*m*	24.77/*m*	23.98/*m*	**23.24/*m***	
	Least value	33.97/*m*	**31.08/*m***	35.14/*m*	41.57/*m*	43.06/*m*	33.27/*m*	

ble of coordinated control among multiple intersections for the time being. As the number of intersections increases, the computational complexity of the model will significantly rise, which may lead to excessive consumption of computational resources. Additionally, the current model's reward function is mainly based on the number of queued vehicles. While this design reflects the bottleneck problem in traffic flow, in more complex traffic scenarios, the queue count is not the sole optimization objective. The focus of the next work will further consider the mutual cooperation and communication among multiple agents, employing multi-agent reinforcement learning methods to optimize large-scale urban traffic networks. At the same time, the design of the reward function will be expanded to integrate more dimensions of traffic factors, such as road throughput and pedestrian signal coordination. Moreover, how to ensure effective coordination between different intersections within a region while reducing the overall computational complexity of the algorithm will be a critical challenge that needs to be addressed in future work.

Author Contributions HZ and ZF is responsible for data generation, data processing, machine learning model design, coding, Interpretation of results, writing of the first draft, and provided critical feedback on the manuscript and methods used in this study. YC and HD and QJ and XZ provided data processing and review of the manuscript.

Funding This work was supported by the Fujian Province National Natural Science under Grant (2021J01319). National Natural Science Foundation of China under Grant no. 61802133.

Data Availability The datasets generated during and/or analyzed during the current study are available from the correspond author.

Declarations

Conflict of interest The authors declare that they have no Conflict of interest.

Ethics approval and consent to participate Not applicable.

Consent for publication: Not applicable.

References

1. Yau KLA, Qadir J, Khoo HL, Ling MH, Komisarczuk P (2017) A survey on reinforcement learning models and algorithms for traffic signal control. ACM Comput Surv (CSUR) 50(3):1. https://doi.org/10.1145/3068287
2. Xiao Z, Tong H, Qu R, Xing H, Luo S, Zhu Z, Song F, Feng L (2023) CapMatch: Semi-supervised contrastive transformer capsule with feature-based knowledge distillation for human activity recognition, IEEE Transactions on Neural Networks and Learning Systems
3. Xiao Z, Xu X, Xing H, Zhao B, Wang X, Song F, Qu R, Feng L (2024) DTCM: Deep Transformer Capsule Mutual Distillation for Multivariate Time Series Classification, IEEE Transactions on Cognitive and Developmental Systems
4. Miletić M, Ivanjko E, Mandžuka S, Nečoska DK (2021) Combining neural gas and reinforcement learning for adaptive traffic signal control. In: 2021 International Symposium ELMAR, IEEE, pp. 179–182. https://doi.org/10.1109/ELMAR52657.2021.9550948
5. Chen Y, Zhang H, Liu M, Ye M, Xie H, Pan Y (2023) Traffic signal optimization control method based on adaptive weighted averaged double deep Q network. Appl Intell 53(15):18333. https://doi.org/10.1007/s10489-023-04469-9
6. Essa M, Sayed T (2020) Self-learning adaptive traffic signal control for real-time safety optimization. Accident Anal Prevent 146:105713
7. Li L, Lv Y, Wang FY (2016) Traffic signal timing via deep reinforcement learning. IEEE/CAA J Automatica Sinica 3(3):247. https://doi.org/10.1109/JAS.2016.7508798
8. Wu C, Kim I, Ma Z (2023) Deep Reinforcement Learning Based Traffic Signal Control: A Comparative Analysis. Proc Comput Sci 220:275
9. Cabrejas-Egea A, Zhang R, Walton N (2021) Reinforcement learning for traffic signal control: comparison with commercial systems. Trans Res Proc 58:638
10. Swapno SMR, Chhabra G, Kaushik K, Nobel SN, Islam MB, Shahiduzzaman M (2023) An Adaptive Traffic Signal Management System Incorporating Reinforcement Learning, in 2023 Annual International Conference on Emerging Research Areas: International Conference on Intelligent Systems (AICERA/ICIS), IEEE, pp. 1–6. https://doi.org/10.1109/AICERA/ICIS59538.2023.10420185
11. VM SM, Krishnendhu S, Mohandas P (2023) Real-Time Traffic Signal Prediction and Control using Deep Q-Network, in 2023 International Conference on Computer, Electronics & Electrical Engineering & their Applications (IC2E3), IEEE, pp. 1–6. https://doi.org/10.1109/IC2E357697.2023.10262818
12. Xu Z, Zhang L, Qi F (2023) Adaptive traffic light control based on reinforcement learning under different stages of autonomy, in 2023 35th Chinese Control and Decision Conference (CCDC), IEEE, pp. 715–720. https://doi.org/10.1109/CCDC58219.2023.10327174
13. Fan L, Yang Y, Ji H, Xiong S (2023) Research on Cooperative Control of Traffic Signals based on Deep Reinforcement Learning. In: 2023 IEEE 12th data driven control and learning systems conference (DDCLS), IEEE, pp. 1608–1612. https://doi.org/10.1109/DDCLS58216.2023.10167232
14. Xu M, Wu J, Huang L, Zhou R, Wang T, Hu D (2020) Network-wide traffic signal control based on the discovery of critical nodes and deep reinforcement learning. J Intell Trans Syst 24(1):1. https://doi.org/10.1080/15472450.2018.1527694
15. Shanmugasundaram P, Sinha A (2021) Intelligent traffic control using double deep q networks for time-varying traffic flows, in 2021 8th International Conference on Signal Processing and Integrated Networks (SPIN), IEEE, pp. 64–69. https://doi.org/10.1109/SPIN52536.2021.9565961

16. Agafonov A, Myasnikov V (2021) Traffic signal control: A double q-learning approach, in 2021 16th Conference on Computer Science and Intelligence Systems (FedCSIS), IEEE, pp. 365–369. https://doi.org/10.15439/2021F109
17. Kumar R, Sharma NVK, Chaurasiya VK (2024) Adaptive traffic light control using deep reinforcement learning technique. Multimedia Tools Appl 83(5):13851. https://doi.org/10.1007/s11042-023-16112-3
18. Mao F, Li Z, Li L (2022) A comparison of deep reinforcement learning models for isolated traffic signal control. IEEE Intell Trans Syst Mag 15(1):160. https://doi.org/10.1109/MITS.2022.3144797
19. Zhang X, Xu X (2023) FP-WDDQN: An improved deep reinforcement learning algorithm for adaptive traffic signal control, in 2023 IEEE International conference on data mining workshops (ICDMW), IEEE, pp. 44–51. https://doi.org/10.1109/ICDMW60847.2023.00015
20. Bouktif S, Cheniki A, Ouni A, El-Sayed H (2021) Traffic signal control based on deep reinforcement learning with simplified state and reward definitions. In: 2021 4th International conference on artificial intelligence and big data (ICAIBD), IEEE, pp. 253–260. https://doi.org/10.1109/ICAIBD51990.2021.9459029
21. Ren A, Zhou D, Feng J et al (2023) Attention mechanism based deep reinforcement learning for traffic signal control. Appl Res Comput 40(02):430
22. Raeis M, Leon-Garcia A (2021) A deep reinforcement learning approach for fair traffic signal control, in 2021 IEEE international intelligent transportation systems conference (ITSC), IEEE, pp. 2512–2518. https://doi.org/10.1109/ITSC48978.2021.9564847
23. Pálos P, Huszák Á (2020) Comparison of q-learning based traffic light control methods and objective functions, in 2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM), IEEE, pp. 1–6. https://doi.org/10.23919/SoftCOM50211.2020.9238290
24. Wang Z, Schaul T, Hessel M, Hasselt H, Lanctot M, Freitas N (2016) Dueling network architectures for deep reinforcement learning, in International conference on machine learning, PMLR, pp. 1995–2003
25. Bhumeka S, Nahar A, Alam T, Sultan SM (2023) 3-Lane Based Traffic Signal Control Using Sequential-Duel Deep Q-Network (SD-DQN), in 2023 IEEE 5th Eurasia Conference on Biomedical Engineering, Healthcare and Sustainability (ECBIOS), IEEE, pp. 112–115. https://doi.org/10.1109/ECBIOS57802.2023.10218510
26. Sahu SP, Dewangan DK, Agrawal A, Priyanka TS (2021) Traffic light cycle control using deep reinforcement technique, in 2021 international conference on artificial intelligence and smart systems (ICAIS), IEEE, pp. 697–702. https://doi.org/10.1109/ICAIS50930.2021.9395880
27. Liang X, Du X, Wang G, Han Z (2019) A deep reinforcement learning network for traffic light cycle control. IEEE Trans Vehicular Technol 68(2):1243. https://doi.org/10.1109/TVT.2018.2890726
28. Gu S, Zhang T, Zhang Y (2023) Inverse Reinforcement Learning Integrated Reinforcement Learning for Single Intersection Traffic Signal Control, in 2023 5th International conference on industrial artificial intelligence (IAI), IEEE, pp. 1–6. https://doi.org/10.1109/IAI59504.2023.10327510
29. Ni W, Wang P, Li Z, Li C (2023) Traffic Signal Control Optimization Based on Deep Reinforcement Learning with Attention Mechanisms, in International conference on neural information processing, Springer, pp. 147–158. https://doi.org/10.1007/978-981-99-8067-3_11
30. Yang D, Zai W, Yan L, Wang J (2023) Low Carbon City Traffic Signal Control Based on Deep Reinforcement Learning, in 2023 Panda Forum on Power and Energy (PandaFPE), IEEE, pp. 1797–1801. https://doi.org/10.1109/PandaFPE57779.2023.10140589
31. Bouktif S, Cheniki A, Ouni A, El-Sayed H (2023) Deep reinforcement learning for traffic signal control with consistent state and reward design approach. Knowl-Based Syst 267:110440
32. Garg D, Chli M, Vogiatzis G (2018) Deep reinforcement learning for autonomous traffic light control, in 2018 3rd IEEE international conference on intelligent transportation engineering (ICITE), IEEE, pp. 214–218. https://doi.org/10.1109/ICITE.2018.8492537,
33. Sun H, Chen C, Liu Q, Zhao J (2019) Traffic signal control method based on deep reinforcement learning. Comput Sci 47(2):169
34. Zhang R, Ishikawa A, Wang W, Striner B, Tonguz OK (2020) Using reinforcement learning with partial vehicle detection for intelligent traffic signal control. IEEE Trans Intell Trans Syst 22(1):404. https://doi.org/10.1109/TITS.2019.2958859
35. Guo M, Wang P, Chan CY, Askary S (2019) A reinforcement learning approach for intelligent traffic signal control at urban intersections, in 2019 IEEE Intelligent Transportation Systems Conference (ITSC), IEEE, pp. 4242–4247. https://doi.org/10.1109/ITSC.2019.8917268
36. Xu M, Wu J, Huang L, Zhou R, Wang T, Hu D (2020) Network-wide traffic signal control based on the discovery of critical nodes and deep reinforcement learning. J Intell Trans Syst 24(1):1. https://doi.org/10.1080/15472450.2018.1527694
37. Egea AC, Howell S, Knutins M, Connaughton C (2020) Assessment of reward functions for reinforcement learning traffic signal control under real-world limitations, in 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, pp. 965–972. https://doi.org/10.1109/SMC42975.2020.9283498
38. Zhong D, Boukerche A (2019) Traffic signal control using deep reinforcement learning with multiple resources of rewards, in Proceedings of the 16th ACM international symposium on performance evaluation of wireless Ad Hoc, sensor, & ubiquitous networks, pp. 23–28. https://doi.org/10.1145/3345860.3361522
39. Lee J, Chung J, Sohn K (2019) Reinforcement learning for joint control of traffic signals in a transportation network. IEEE Trans Vehicular Technol 69(2):1375. https://doi.org/10.1109/TVT.2019.2962514
40. Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks, in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7132–7141
41. Arulkumaran K, Deisenroth MP, Brundage M, Bharath AA (2017) Deep reinforcement learning: a brief survey. IEEE Signal Process Mag 34(6):26. https://doi.org/10.1109/MSP.2017.2743240
42. Zheng Y, Hao JY, Zhang ZZ, Meng ZP, Hao XT (2020) Efficient multiagent policy optimization based on weighted estimators in stochastic cooperative environments. J Comput Sci Technol 35:268. https://doi.org/10.1007/s11390-020-9967-6