Digital Transportation and Safety

ARTICLE

Open Access

https://doi.org/10.48130/dts-0026-0004

Digital Transportation and Safety **2026**, 5(1): 42–52

Double deep network-based traffic signal optimization method for isolated intersections

Rongjian Dai* and Yanzhen Li

School of Qilu Transportation, Shandong University, Jinan, Shandong 250100, China

* Correspondence: rjdai@sdu.edu.cn (Dai R)

Abstract

This study addresses the limitations of existing reinforcement learning (RL)-based traffic signal control methods, which typically optimize either the signal phase sequence or phase duration independently. We propose a novel joint optimization framework based on the Double Deep Q-Network (DDQN) that simultaneously determines both the phase sequence and phase duration. To ensure stability, the base phase duration is determined using the classical Webster method. Furthermore, a hybrid state representation is developed by integrating both microscopic and macroscopic traffic features, such as queue length and vehicle delay. A Squeeze-and-Excitation (SE) attention mechanism is introduced to guide the agent's attention toward critical traffic attributes. Simulation experiments conducted on the SUMO platform demonstrate that the proposed method significantly reduces average queue length and vehicle travel time when compared to traditional fixed-time and vehicle-actuated control strategies, particularly under medium to high traffic demand. The results validate the effectiveness, robustness, and practical applicability of the method for intelligent signal control in complex urban intersections.

Keywords: Traffic signal control, Reinforcement learning, Double Deep Q-Network, Phase sequence, Phase duration

Citation: Dai R, Li Y. 2026. Double deep network-based traffic signal optimization method for isolated intersections. *Digital Transportation and Safety* 5(1): 42–52 https://doi.org/10.48130/dts-0026-0004

Introduction

With the accelerating pace of urbanization, traffic congestion has emerged as a major bottleneck impeding sustainable urban development. It not only reduces the efficiency of road resource utilization and prolongs commuting times, but also contributes to a range of environmental pollution issues[1]. To address these challenges, researchers have proposed a variety of traffic management strategies, including signal timing optimization, roadway expansion, and the implementation of intelligent transportation systems[2,3]. Among these, traffic signal control has gained prominence as a core technology for enhancing network efficiency, owing to its relatively low deployment cost and short implementation cycle.

With continuous advancements in traffic data acquisition technologies, traffic signal control methods have undergone significant evolution and innovation[4,5]. In general, their development can be divided into three stages: static control based on fixed-time plans, vehicle-actuated control triggered by detectors[6,7], and adaptive control based on real-time optimization[8,9]. The latter two categories leverage sensor data to capture real-time traffic volumes and dynamically adjust signal timing plans[10]. Although these methods have improved traffic efficiency to some extent, they still suffer from shortcomings such as delayed response and limited generalization ability in dealing with the highly nonlinear and stochastic characteristics of urban traffic flow.

In recent years, the rapid advancement of artificial intelligence has introduced new research methods and technical paradigms to the field of traffic signal control[11]. Among them, reinforcement learning (RL) has become a research hotspot due to its ability to make autonomous decisions and optimize policies in dynamic environments[12–14]. In RL, an agent interacts with the environment by trial and error, continuously learning and updating the mapping between actions and states—namely, the signal control policy—and adaptively adjusts signal plans based on the long-term cumulative reward (Q-value). However, when handling high-dimensional continuous state spaces, traditional RL algorithms exhibit insufficient representational capacity, resulting in poor control performance in complex network environments[15,16].

Deep reinforcement learning (DRL), which combines the perceptual strength of deep learning (DL) with the sequential decision-making capabilities of RL, offers a promising solution to these limitations[17–19]. DRL agents follow a model-free learning paradigm, leveraging the powerful perception capabilities of deep networks to learn fine-grained control strategies directly from traffic data, thereby enabling more flexible and adaptive signal control schemes[20]. As a result, DRL-based traffic signal control approaches have attracted growing interest from the academic community[21–23].

At present, reinforcement learning (RL)-based traffic signal control (TSC) strategies can be broadly divided into two categories: one focuses on optimizing the phase duration under a fixed phase sequence (i.e., timing optimization); the other focuses on dynamically adjusting the phase sequence under fixed or preset phase durations (i.e., phase sequence optimization). In timing optimization strategies, the agent adjusts the green duration of each signal phase based on real-time traffic demand to improve intersection efficiency[24]. This approach is adaptable to a certain extent and can respond to traffic flow fluctuations. However, its flexibility and adaptability are significantly limited under peak or high-density traffic conditions[25]. Moreover, due to the highly uneven arrival patterns across directions, the performance of timing-based strategies often lacks consistency and robustness in practical applications. In contrast, phase sequence optimization strategies offer greater flexibility and responsiveness. These strategies allow the agent to dynamically select the optimal phase according to real-time traffic

conditions. When the same phase is consecutively selected, the clearance interval can be skipped, thereby effectively extending green time in a specific direction and improving traffic flow stability[26–28]. Typically, such strategies switch phases at fixed intervals, enabling them to adapt to varying traffic patterns throughout the day. Despite their advantages, one critical challenge lies in determining the appropriate phase-switching interval[29]. If the interval is too short, frequent phase transitions may lead to excessive yellow time and reduced intersection capacity, as well as increased computational burden during training and operation. On the other hand, excessively long intervals may reduce the system's ability to promptly respond to traffic fluctuations, degrade real-time performance, and hinder the learning process, leading to suboptimal policies.

Reinforcement learning (RL) has attracted increasing attention in traffic signal control, yet its practical application remains constrained by several key challenges. A central difficulty is that simultaneously optimizing phase sequence and phase duration dramatically enlarges the action space, particularly at intersections with multiple phases. This high dimensionality increases exploration complexity and often leads to slow or unstable convergence. As a result, many existing studies restrict optimization to either phase duration or phase sequence, thereby simplifying the control problem at the cost of reduced operational flexibility. This separation limits the controller's ability to cope with heterogeneous demands and rapidly fluctuating traffic conditions. Moreover, conventional RL-based controllers typically rely on coarse-grained state representations that lack lane-level detail. Without sufficient microscopic information, these models struggle to identify localized congestion phenomena such as queue spillback or uneven lane utilization, ultimately reducing their responsiveness in complex real-world environments.

To address these limitations, this study proposes a DDQN-based control framework that incorporates a structured action design, enabling the integrated optimization of phase sequence and phase duration while avoiding uncontrolled action-space inflation. A multi-scale state representation is constructed to capture both macroscopic and microscopic traffic features, and a Squeeze-and-Excitation (SE) attention mechanism is introduced to emphasize congestion-critical dimensions within the state vector. These components enhance the stability of the learning process, improve convergence efficiency, and support more flexible and adaptive control under dynamic urban traffic conditions.

Motivated by these shortcomings, this study proposes an enhanced Double Deep Q-Network (DDQN) framework that supports integrated decision-making for both phase sequence and phase duration. The major contributions are as follows:

(1) Unified decision-making structure: The proposed framework combines phase selection and timing adjustment within a single decision architecture, improving flexibility and adaptability under dynamic traffic conditions.

(2) Domain-informed phase duration anchoring: A baseline phase duration derived from the classical Webster method is incorporated to stabilize control actions and prevent excessively frequent or erratic switching, thereby improving policy robustness.

(3) Multi-scale state representation with SE attention: The state vector integrates macroscopic indicators (e.g., average queue length, delay) with microscopic lane-level features (e.g., vehicle counts), while the SE module selectively emphasizes critical congestion features to enhance decision relevance and accuracy.

Background and problem description

This section introduces the fundamental concepts of RL and DRL, followed by the RL-based traffic signal control framework for intersections.

Reinforcement learning and deep reinforcement learning

Reinforcement learning (RL) is a learning-based control methodology aimed at maximizing long-term cumulative rewards. It has been widely applied in optimization problems that require sequential decision-making in dynamic environments. As a major subfield of machine learning, RL enables an agent to continuously interact with its environment and iteratively refining its behavior through trial and error, ultimately learning an optimal control policy.

Formally, an RL problem is typically modeled within the framework of a Markov Decision Process (MDP), defined by a five-element tuple $\langle S, A, R, P, \gamma \rangle$. The components are described as follows:

State space S: A finite set of all possible environment states. The agent perceives the current state to make decisions. If the Markov property is satisfied—i.e., the current state contains all relevant information for future decision making—then the agent can ignore historical states and rely solely on the current state for policy selection.

Action space A: A set of all valid actions that the agent can execute in a given state s_t. At each time step t, the agent selects an action a_t from the action space according to a policy π, aiming to maximize its long-term cumulative return.

Transition probability P: The transition function $P(s_{t+1}|s_t,a_t)$ defines the conditional probability of transitioning to state s_{t+1} after taking action a_t in state s_t. This models the dynamics of the environment.

Reward function R: The reward function $R(s_t,a_t)$ specifies the immediate reward r_t received after performing action a_t in state s_t. This reward signal is critical for evaluating the quality of actions and guiding policy updates.

The expected return G_t is defined as the discounted cumulative reward from time step t onward, as shown in Eq. (1):

$$G_t = R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \cdots = \sum_{k=0}^{n} \gamma^k R_{t+k} \tag{1}$$

where, $\gamma \in [0,1]$ is the discount factor, used to balance the importance of immediate and future rewards. A smaller γ favors short-term gain, while a larger γ emphasizes long-term return.

Based on the MDP framework, the goal of an agent is to learn an optimal policy π^* that maximizes the expected cumulative return for any given state. In this study, we employ Q-learning, a widely used value-based RL algorithm. In Q-learning, the agent updates the action-value function (i.e., Q-function) through continuous interaction with the environment. The Q-value reflects the expected return of taking action a in state s, and is updated iteratively as follows:

$$Q_{t+1}(s,a) = Q_t(s,a) + \alpha \delta_{t+1} \tag{2}$$

Here, $\alpha \in [0,1]$ is the learning rate, and δ_{t+1} is the temporal-difference (TD) error, defined as:

$$\delta_{t+1}(s,a) = R_t + \gamma \max_{a_{t+1} \in A} Q_t(s_{t+1}, a_{t+1}) - Q_t(s,a) = y_t - Q_t(s,a) \tag{3}$$

The TD target is then given by:

$$y_t = R_t + \gamma \max_{a_{t+1} \in A} Q_t(s_{t+1}, a_{t+1}) \tag{4}$$

To approximate Q-values in high-dimensional state-action spaces, the Deep Q-Network (DQN) algorithm uses a neural network as a nonlinear function approximator. The DQN typically includes an

input layer to receive state features, multiple hidden layers for feature extraction, and an output layer to estimate Q-values for all available actions. The network is parameterized by $Q(s,a;\theta)$, and denoted as $e_t = (s_t,a_t,R_t,s_{t+1})$.

To improve training stability, DQN incorporates an experience replay mechanism. At each time step t, the agent stores the interaction tuple $e_t = (s_t,a_t,R_t,s_{t+1})$ into an experience replay buffer. Subsequently, during each model update, a mini-batch of samples is randomly drawn from the experience replay buffer. This approach effectively breaks the temporal correlations in the data while enhancing the independence and diversity of training samples.

During the training process, the DQN employs the same TD target as standard Q-learning, as shown in Eq. (5):

$$y_t^{\mathrm{DQN}} = R_t + \gamma \max_{a_{t+1}\in A} Q_t(s_{t+1},a_{t+1};\theta) \tag{5}$$

However, during the process of Q-value updating, using the same network for both action selection and evaluation can lead to overestimation of Q-values. In other words, this approach tends to overestimate the value of certain actions, thereby affecting the quality of the learned policy and reducing training stability. To address this issue, Hasselt et al. proposed the Double Network Mechanism[30], known as DDQN. This method decouples action selection from value evaluation, effectively reducing the estimation bias in Q-values. In DDQN, the current network with parameter θ is used to select the action, while a separate target network with parameter θ^- is introduced to evaluate the value of the selected action, as illustrated in Fig. 1. The corresponding TD target is given by Eq. (6):

$$y_t^{\mathrm{DDQN}} = R_t + \gamma Q_t\left(s_{t+1}, \arg\max_{a_{t+1}\in A} Q_t(s_{t+1},a_{t+1};\theta);\theta^-\right) \tag{6}$$

Traffic signal control modeling

In the framework of RL, traffic signal control (TSC) can be modeled as an agent–environment interaction process. The decision-making environment evolves continuously due to the highly dynamic and uncertain nature of traffic flows. These changes stem from temporal, spatial, and behavioral variations in traffic patterns. As a result, the agent must constantly learn and refine its decision-making policy to consistently make optimal action choices at every time step. This ongoing policy learning process enables the agent to accumulate experience through interaction with the environment, recognize emerging traffic state patterns, and adjust its control behavior in a timely manner. Such adaptability is essential for maintaining responsiveness and control efficiency in complex traffic scenarios. Especially in real-world systems, fixed control strategies often fail to accommodate all possible traffic conditions. The RL framework offers an effective mechanism for continuously iterating and improving policies during both training and deployment, thereby adapting to environmental changes.

In the RL-based intersection traffic signal control framework proposed in this study, the intersection is modeled as the environment E, and the agent G interacts with it. At each time step t, the agent receives the current environment state s_t, selects an action a_t, and obtains an immediate reward r_t from the environment. The goal of the agent is to learn and execute the optimal action for each perceivable state, that is, to select the appropriate traffic phase to optimize the defined control objectives.

As shown in Fig. 2, this study considers a typical four-arm intersection. Right-turn movements are not controlled by traffic signals and are generally allowed to proceed freely when yielding conditions are met. Therefore, in the optimization of traffic signal control strategies, only eight signal-controlled traffic movements are considered: Westbound Through (WBT), Westbound Left (WBL), Eastbound Through (EBT), Eastbound Left (EBL), Southbound Through (SBT), Southbound Left (SBL), Northbound Through (NBT), and Northbound Left (NBL).

This study adopts a four-phase signal control scheme, as illustrated in Fig. 3, where each signal phase corresponds to a set of non-conflicting traffic movement directions, ensuring both safety and operational efficiency at the intersection. For each phase, a base duration g^b is defined—this represents the minimum time that the phase must remain active to maintain traffic flow stability. During this base duration, the signal remains fixed and does not switch. Once the base duration of the current phase expires, the agent evaluates the current traffic state—such as queue length and vehicle waiting time—and determines whether to switch phases. If a switch is deemed necessary, the agent selects the next phase to be executed, thereby realizing dynamic phase sequence optimization. Additionally, if the agent chooses not to switch, the current phase will continue to be extended, enabling joint optimization of phase sequence and phase duration. This mechanism significantly enhances the flexibility of the signal control strategy and improves its responsiveness to complex, real-time traffic variations. For example, if the current phase is Phase 1, the agent can, at the end of its base duration, flexibly choose to either continue with Phase 1 or switch to another phase, as shown in Fig. 3.

This mechanism significantly enhances the flexibility of the signal control strategy and its real-time responsiveness to complex traffic conditions. It is important to note that, to ensure safety during phase transitions, the system must insert a clearance interval each time the signal changes. This interval includes both the yellow light

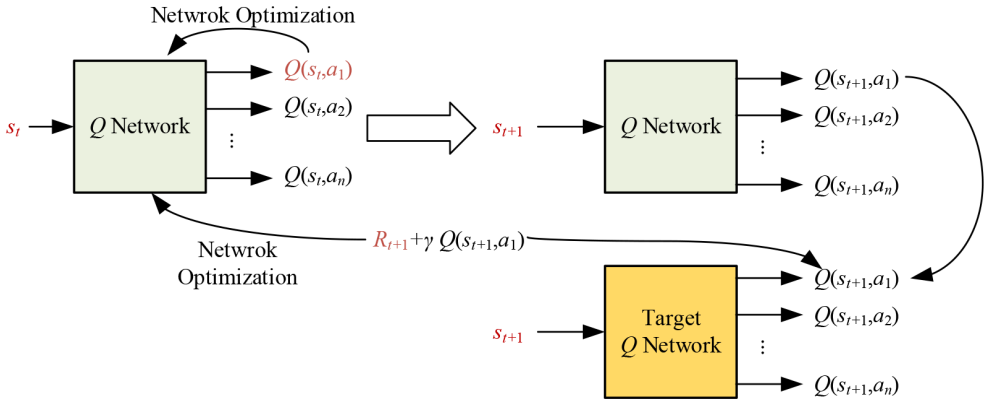

Fig. 1 The update process of DDQN.

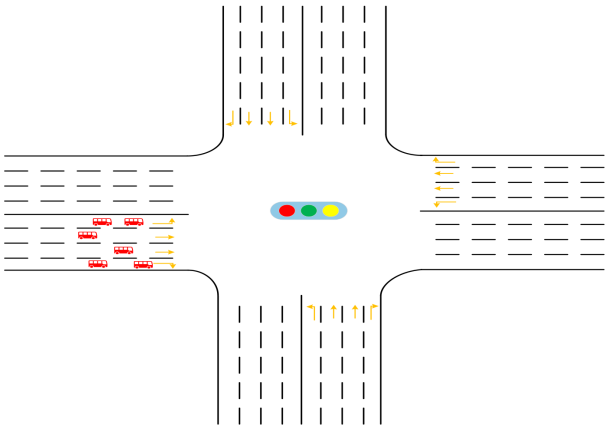

Fig. 2 Typical four-arm signalized intersection.

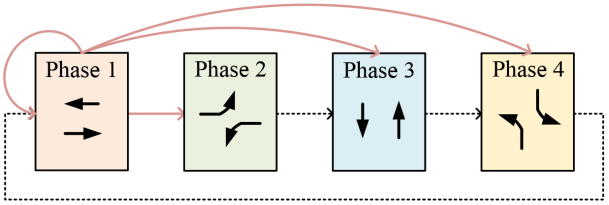

Fig. 3 Signal phase setting.

duration and an all-red period, allowing vehicles from the previous phase to completely clear the intersection before the next phase begins. The inclusion of this safety buffer is essential for improving the operational stability of the signal system and ensuring overall traffic safety.

Traffic signal control based on DDQN

Agent architecture

In this study, we develop a traffic signal control framework based on the DDQN to better adapt to the complexity and variability of urban traffic environments. In the DDQN architecture, the value network and target network work collaboratively to decouple the processes of action selection and evaluation, effectively mitigating the Q-value overestimation problem commonly found in traditional DQN approaches.

As illustrated in Fig. 4, action selection is performed by the value (online) network, while action evaluation is handled by the target network. This separation enables more stable and efficient policy learning, making DDQN particularly suitable for highly dynamic and complex scenarios such as traffic signal control.

The value network takes the observed state s_t as input, processes it through several fully connected (FC) layers to extract features, and outputs the Q-values corresponding to all possible actions under the current state. The parameter of the value network (i.e., θ) is optimized by minimizing the mean squared error (MSE) loss between predicted Q-values and the TD target, as shown in Eq. (7), thereby continuously improving the agent's decision policy.

$$L(\theta) = \mathbb{E}_{(s_t,a_t,R_t,s_{t+1})}\left[\left(y_t^{DDQN} - Q(s_t,a_t;\theta)\right)^2\right] \tag{7}$$

The target network does not directly participate in policy decision-making. It is solely used to compute the next-step Q-value in the TD target. To ensure training stability, the parameter of the target network θ^- is not updated at every time step. Instead, they are periodically synchronized by copying the parameter θ from the value network at fixed intervals. This delayed update mechanism

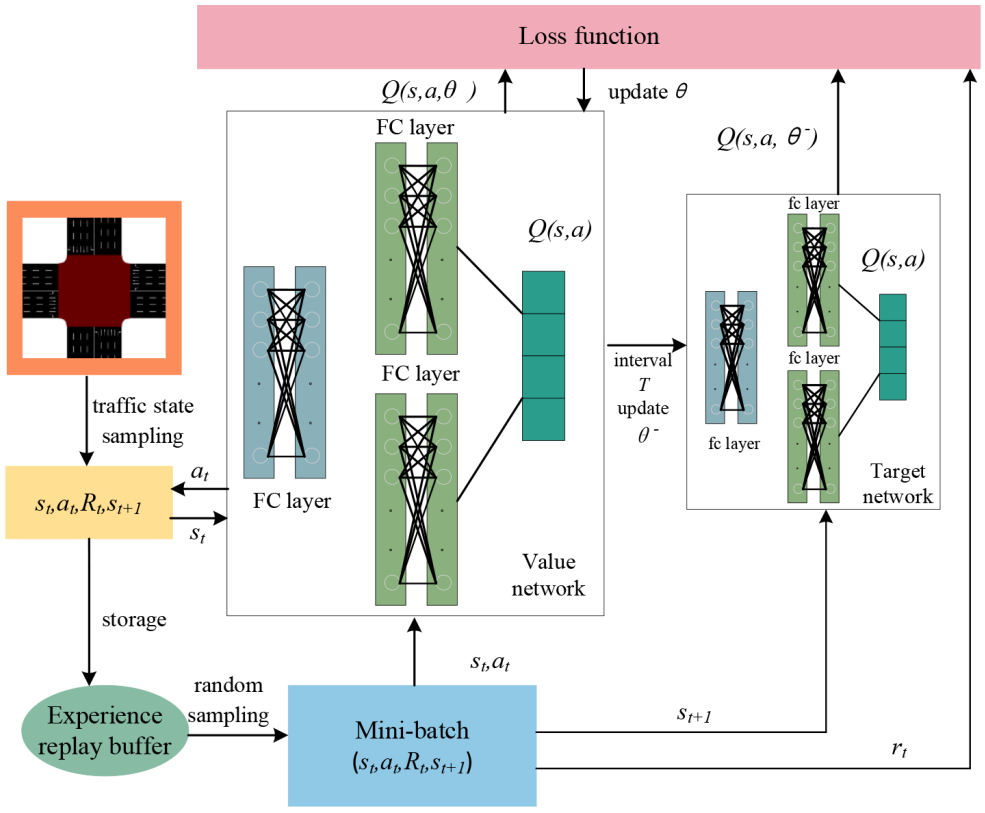

Fig. 4 Traffic signal control framework based on DDQN.

reduces fluctuations in target values during training, allowing the target network to serve as a relatively 'stable reference' over short time horizons. As a result, it improves the convergence behavior of the training process and enhances the overall quality of the learned policy.

In this study, we further optimize the DDQN architecture to enhance training stability and convergence speed. First, for network weight initialization, we adopt a uniform distribution bound by the square root of the reciprocal of the input dimension, i.e., $\sqrt{1/\text{input}}$, to initialize the weights and biases of the first and second fully connected layers. This strategy helps reduce the risk of gradient vanishing or explosion, and ensures more stable model training. Second, we replace the standard ReLU activation with LeakyReLU, which mitigates the 'dying neuron' problem by maintaining gradient flow in the negative domain, thereby improving the robustness of the training process.

To enhance the model's ability to recognize traffic-critical features, the Squeeze-and-Excitation (SE) attention mechanism is incorporated into the DDQN framework as a lightweight feature-refinement module. Rather than altering the reinforcement learning architecture, the SE module adaptively reweighs the state variables to emphasize those most relevant to signal control, such as long queues, low speeds, and prolonged delays—thereby improving the robustness and responsiveness of the learned policy.

As illustrated in Fig. 5, the SE module operates on the input state vectors s and performs channel-wise significance modeling through three sequential steps:

(1) Squeeze: A global average pooling (GAP) operation is applied to s to obtain a compact statistical descriptor z, which captures the aggregated traffic state information:

$$z = GAP(s) \tag{8}$$

(2) Excitation: The descriptor z is passed through two fully connected layers with ReLU and Sigmoid activations to generate a normalized set of importance weights:

$$w = \sigma(W_2\delta(W_1 z)) \tag{9}$$

(3) Reweighting: The generated importance vector w is applied to the original state vector through element-wise multiplication to produce the attention-enhanced state representation s':

$$s' = s \odot w \tag{10}$$

This reweighed state amplifies congestion-critical variables while suppressing less informative ones, yielding a more discriminative input for the DDQN network.

It should be emphasized that the SE mechanism in this study serves as an auxiliary enhancement rather than a standalone performance-determining component. Its role is to refine feature importance within the state vector, not to independently drive control performance. For this reason, and because SE does not modify the action space or decision architecture, a separate ablation experiment is not essential for validating the core contribution of the proposed framework. The influence of the SE module is reflected in its ability to highlight features associated with severe congestion, thereby improving interpretability and contributing to more targeted control decisions. Nonetheless, future research may incorporate quantitative ablation analyses to more systematically examine the marginal contribution of the SE mechanism across different traffic scenarios.

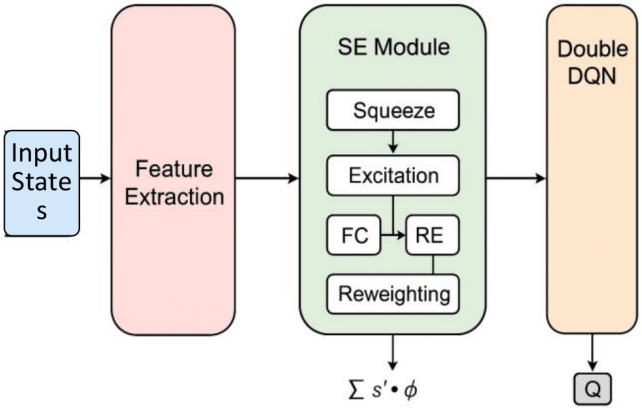

Fig. 5 Network architecture of the SE attention mechanism.

As illustrated in Fig. 4, during training, the agent continuously interacts with the simulation environment. At each time step t, the agent observes the current traffic state s_t, and selects an action a_t according to its current policy. After the action is executed, the environment transitions to a new state s_{t+1}, and returns an immediate reward R_t. The experience tuple (s_t, a_t, R_t, s_{t+1}) is then stored in the experience replay buffer.

To enhance the diversity and independence of training samples, the DDQN framework adopts an experience replay mechanism[31]. During each policy update iteration, a mini-batch of samples is randomly drawn from the replay buffer and used to train the neural network. This random sampling helps break the temporal correlations among samples and prevents instability in the learning process caused by sequential data dependencies.

The detailed implementation process is shown in Table 1.

Phase base duration

The phase base duration is a critical parameter that directly influences the performance of traffic signal control. It governs both the responsiveness and stability of the control strategy. Appropriately determining the base duration is essential for developing

Table 1. Signal control algorithm based on DDQN.

Input: Discount factor γ; Learning rate α; Number of update iterations I; Target network update interval κ; Mini-batch size b; Number of training episodes N; Number of simulation steps T.
Output: Optimized network weights θ^*.

Initialization:
Initialize value network $Q(\cdot;\theta)$ and target network $Q(\cdot;\theta^-)$, and set $\theta^- \leftarrow \theta$;
Initialize experience replay buffer D;
Initialize update counter $Counter \leftarrow 0$.
Detailed algorithm flow:
For episode n = 1 to N:
  Observe initial environment state s_0 and take initial action a_0;
  For t = 1, 2, …, T:
    Observe current state s_t, and select action a_t based on ε-greedy policy;
    Execute action a_t, observe next state s_{t+1}, and receive reward R_t;
    Compute TD value: $TD = y_t^{DDQN} - Q_t(s_t, a_t; \theta_t)$;
    Store experience tuple $\langle \delta_{t+1}, (s_t, a_t, R_t, s_{t+1}) \rangle$ in D.
  End for
  If $n \geq 2$: // skip early episodes to stabilize exploration
    For i = 1 to I:
      Sample a mini-batch B of size b from D;
      Compute TD target y_t using Eq. (6) with the target network;
      Update θ by minimizing loss via gradient descent: $\theta \leftarrow \theta - \alpha \nabla L(\theta)$;
      Increment counter: $Counter \leftarrow Counter + 1$.
      If Counter mod κ== 0:
        Update target network parameters: $\theta^- \leftarrow \theta$.
      End if
    End for
  End if
End for

a signal control policy that is efficient, stable, and adaptive to real-time traffic dynamics.

In this study, the classical Webster method[4] is first employed to calculate the optimal cycle length for the fixed-time signal control scheme. The resulting cycle length serves as a stable baseline for the intersection and prevents the signal plan from operating with excessively short or unstable cycles. It should be noted that Webster's formula is used only to establish the baseline cycle and does not directly determine the phase durations in our framework. Based on this baseline cycle, the initial effective green times for each phase are obtained using an equal allocation principle. Subsequently, the proposed DDQN-based controller dynamically adjusts both the phase sequence and phase duration in real time according to the observed traffic states. This design ensures that the controller remains effective even when actual traffic conditions deviate from the simplified assumptions underlying the Webster method.

Thus, the Webster-derived cycle acts only as an initialization reference to prevent excessively short or unstable cycles, while the RL agent determines the actual timing during operation. This design ensures that the system remains fully adaptive and maintains satisfactory performance even when traffic conditions deviate from the assumptions inherent in the Webster method.

According to the Webster method, the optimal signal cycle length C can be calculated using Eq. (11). L denotes the total lost time per cycle, which includes both yellow and all-red intervals. Y denotes the sum of the critical flow ratios across all phases, calculated as shown in Eq. (12). The critical flow ratio y_i for phase i is denoted as the ratio of the critical lane flow q_i to the saturation flow ϑ_i of that phase, as expressed in Eq. (13).

$$C = \frac{1.5L + 5}{1 - Y} \tag{11}$$

$$Y = \sum_{i=1}^{4} y_i \tag{12}$$

$$y_i = \frac{q_i}{\vartheta_i} \tag{13}$$

In this study, the effective green time under equal allocation is adopted as a reference, as shown in Eq. (14):

$$g^b \leftarrow \frac{C - L}{4} \tag{14}$$

State space

In the RL framework, the state space serves as the foundation for the agent to perceive the environment and make decisions. The quality of the state representation directly affects both the training efficiency and the generalization capability of the learned policy. For traffic signal control tasks, the state space should comprehensively reflect the real-time operational status of the intersection.

In this study, a lane-group-based state representation is adopted instead of a fine-grained lane-level description. This approach preserves essential traffic characteristics while avoiding unnecessary growth in state dimensionality. As defined in Eqs (15)–(17), the state consists of traffic-flow attributes and signal-phase information.

For each lane group p, the traffic-related variables include the average queue length L_p, average speed V_p, average waiting time W_p, and the number of vehicles N_p, jointly reflecting the congestion and movement conditions of the approach. In addition to these traffic indicators, the state also incorporates the current and previous signal phases (i.e., φ_t and φ_{t-1}), encoded in a one-hot format to capture phase transitions.

$$s_t = \left\{ s_t^{tra}, s_t^{sig} \right\} \tag{15}$$

$$s_t^{tra} = \left\{ L_p, V_p, W_p, N_p \right\}_{p=1,2,3,4} \tag{16}$$

$$s_t^{sig} = \{ \varphi_t, \varphi_{t-1} \} \tag{17}$$

To enhance the model's perception capability, we constructed a hybrid multi-scale representation that integrates both microscopic (queue length L_p, vehicle count N_p) and macroscopic (delay W_p, speed V_p) features at the lane-group level. All variables are normalized to the range [0,1] using min-max scaling, and the complete state vector is formed by concatenating the normalized traffic features with the phase encodings.

This unified representation enables the agent to accurately capture congestion patterns and temporal variations in traffic flow, providing a compact yet informative basis for effective signal control decision-making.

Action space

The design of the action space reflects the set of control operations available to the agent. In this study, the action space A is defined as the set of available signal phases, as shown in Eq. (18):

$$A = \{ p_1, p_2, p_3, p_4 \} \tag{18}$$

Here, p_1 corresponds to executing phase 1 (as illustrated in Fig. 3), while p_2, p_3, and p_4 correspond to the remaining three phases. This design allows the agent to flexibly adjust the phase sequence during the control process.

In this study, each phase is assigned the same base duration g^b. Once the base duration of the current phase ends, the agent selects the next phase to execute from the action set A. If the agent chooses to continue the current phase, the green time is extended without requiring any clearance time. However, if the agent opts to switch to a different phase, a clearance interval (including yellow and all-red times) must be executed before the new phase begins. In addition, when no vehicles are present at the intersection, the agent will automatically select the major phase, i.e., the one corresponding to the highest traffic demand.

Reward function

The reward function plays a central role in guiding the learning behavior of reinforcement learning (RL) models. In this study, queue length is selected as the sole reward indicator because it maintains a clear and direct relationship with congestion levels and can be reliably measured in both simulation environments and real-world deployment. Compared with delay or stop frequency, queue length exhibits lower temporal volatility and stronger numerical stability, which helps reduce reward noise and enhances convergence for value-based RL methods. Moreover, using a single, physically interpretable metric avoids the complexity and scale inconsistencies inherent in multi-objective reward formulations, which often require careful weighting or normalization across heterogeneous indicators.

At each time step t, the reward is defined as:

$$R_t = -\left(\frac{1}{N_{lane}} \sum_{l=1}^{N_{lane}} Q_l^t \right) \tag{19}$$

where, N_{lane} is the total number of entry lanes at the intersection (excluding dedicated right-turn lanes); Q_l^t represents the number of queued vehicles on lane l at time step t.

To reflect performance trends, a reward difference metric is also defined as:

$$\Delta R_t = R_t - R_{t-1} \tag{20}$$

A positive value $\Delta R_t > 0$ indicates improved traffic flow, evidenced by a reduction in queue length under the current policy.

While queue length offers a stable and consistent learning signal, we acknowledge that traffic signal control is inherently multi-dimensional, involving factors such as delay, stop frequency, fuel consumption, and emissions. Extending the reward to a multi-objective or weighted formulation, therefore, represents a promising direction for future research.

It is also important to emphasize that the reward function remains unchanged across all traffic demand levels. Instead, the proposed DDQN-SE controller adapts to varying traffic conditions through its real-time state perception and continuous interaction with the environment. This enables the learned policy to generalize effectively across low-, medium-, and high-demand scenarios without requiring scenario-specific modifications to the reward design.

Simulation experiments and results

Scenario setup

In this study, simulation experiments are conducted based on a typical four-approach intersection, as illustrated in Fig. 2. Each approach includes four lanes: the innermost lane is dedicated for left turns, the outermost lane is for right turns, and the two middle lanes serve as through lanes. The baseline traffic demand for each direction is provided in Table 2. The control zone extends 250 m upstream from the intersection, and the speed limit is set to 13.89 m/s. The clearance time (including yellow and all-red intervals) is fixed at 5 s, and the phase base duration is set to 12 s for all phases.

The intersection environment is constructed using SUMO (Simulation of Urban MObility), a microscopic traffic simulation platform. SUMO enables the modeling of intersections under various traffic scenarios and provides the flexibility to integrate external control algorithms. Based on this platform, the proposed DDQN-based traffic signal control algorithm is implemented and evaluated.

The training parameters of the DDQN model are summarized in Table 3. An ε-greedy exploration strategy is employed to balance exploration and exploitation during training. The initial exploration rate is set to 1.0, and it is gradually annealed to 0.02 over the course of training. This ensures that the agent performs sufficient exploration during the early stages, and gradually shifts toward exploiting the learned optimal policy in later stages.

Table 2. Basic traffic demand.

Entry approach	Through (pcu/h)	Left-turn (pcu/h)
Westbound approach	400	100
Northbound approach	200	100
Eastbound approach	380	180
Southbound approach	200	150

To comprehensively evaluate the performance of the proposed traffic signal control method under various scenarios, two key metrics are selected: average travel time, and average queue length. Average travel time serves as a core indicator of intersection efficiency—shorter values indicate higher overall traffic throughput and improved flow. Average queue length, defined as the number of vehicles in a lane moving at speeds less than 0.1 m/s, reflects the level of congestion at the intersection. A longer queue length typically signals a delayed response of the control strategy to evolving traffic conditions.

This study focuses on developing a reinforcement learning-based traffic signal control method capable of jointly optimizing phase sequence and phase duration. Since most existing RL approaches optimize only one of these aspects and rely on different action definitions or phase structures, direct comparison requires substantial modifications to their architectures. Therefore, the experimental evaluation concentrates on fixed-time and vehicle-actuated controllers, which represent the most widely implemented strategies in practice, and provide a clear benchmark for assessing the improvements achieved by the proposed integrated optimization framework. Exploring the integration of the proposed design into policy-gradient or actor-critic RL methods, such as PPO, will be considered in future work.

Results and analysis

Figure 6 illustrates the variation in average queue lengths across different traffic movements corresponding to each signal phase over the course of DDQN training. It can be observed that, as training progresses, the average queue lengths for all directions exhibit a steady downward trend, indicating that the agent gradually learns an effective signal control strategy. After approximately 400 training episodes, the queue length curves for all phases begin to stabilize, demonstrating good convergence behavior. This confirms the stability and reliability of the proposed control method, thereby validating the effectiveness and feasibility of the DDQN-based traffic signal control strategy in urban intersection scenarios.

It is also noteworthy that the final converged queue lengths differ among phases and show a positive correlation with the traffic volumes of their corresponding directions. This outcome can be attributed to the design of the reward function, which applies equal weights across all directions without prioritizing higher-demand flows. While this equal-weighting strategy ensures fairness, it may limit the model's ability to strengthen control over dominant directions under imbalanced traffic conditions.

To validate the feasibility of the phase base duration determination method, the performance of the DDQN-based control method was tested under different phase base durations, as shown in Fig. 7. When the base duration is set to 8 s, the average travel time reaches its highest value, exceeding 80 s. At this point, signal phase switching occurs too frequently, and a significant amount of time is spent

Table 3. Value used for training parameters.

Parameters	Value	Description
Total training episodes	1,500	Total number of training episodes during which the agent interacts with the environment and updates its policy
Maximum simulation steps per episode	3,600	The maximum number of simulation steps executed in a single training episode
Target network update interval κ	3	The target network is updated once every three updates of the main (evaluation) network
Batch size	64	The number of samples used in each training batch for network parameter updates
Learning rate α	0.0025	The learning rate used for training the DDQN network
Discount factor γ	0.95	The discount factor used to calculate the cumulative future reward

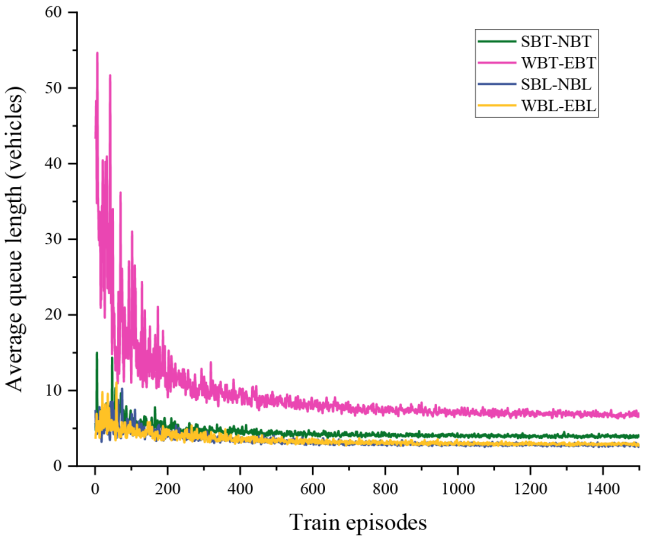

Fig. 6 The trend of average queue length variation for each traffic flow direction.

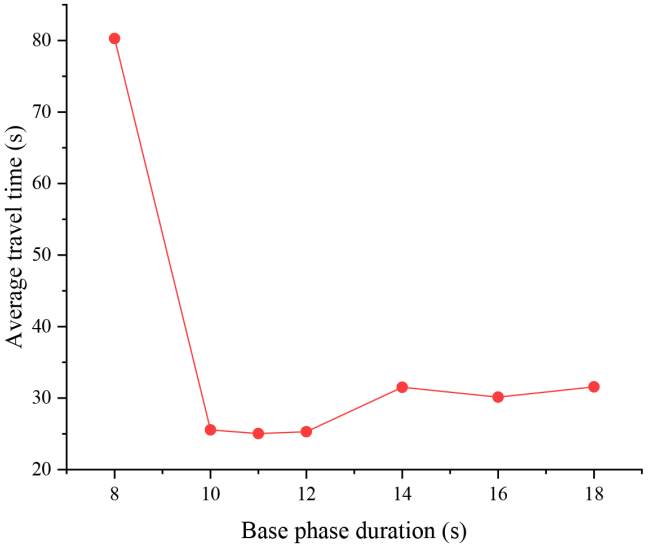

Fig. 7 Impact of base phase duration on the average travel time at the intersection.

on clearance phases, which severely reduces the intersection's throughput. As the base duration increases, the average travel time decreases significantly, reaching its optimal range between 10 and 12 s, with an average value stabilizing around 27 to 28 s. However, when the base duration exceeds 14 s, the average travel time starts to gradually increase. This indicates that while a larger base duration reduces the frequency of phase switches, it also decreases the system's responsiveness to changing traffic conditions. In summary, a base duration range of 10 to 12 s is identified as the optimal setting for the current traffic scenario. This range strikes a good balance between signal switching costs and response speed. The results validate the rationality of the phase base duration setting method proposed in this study.

To validate the performance of the proposed DDQN-based traffic signal control method, comparison experiments were designed against fixed-time control and vehicle-actuated signal control methods. Figure 8 presents a comparison of the average queue length across different traffic directions under the three control

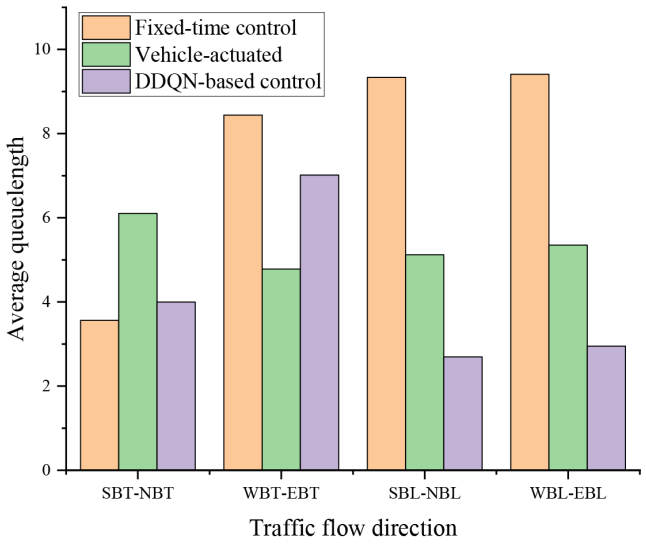

Fig. 8 Comparison of directional average queue lengths under three traffic signal control strategies.

strategies. It is observed that the DDQN-based control strategy outperforms the other strategies in most traffic directions, demonstrating superior queue control performance. Specifically, in the north-south left turn (SBL-NBL) and east-west left turn (WBL-EBL) directions, the DDQN-based control strategy effectively keeps the queue length under three vehicles, significantly outperforming both fixed-time control and vehicle-actuated control. However, a special case is observed in the east-west through (WBT-EBT) direction, where the queue length under the DDQN-based control strategy is higher than that of the vehicle-actuated control. This phenomenon can be attributed to the design of the current reward function, which does not incorporate differentiated weighting for high-traffic directions. As a result, the agent did not sufficiently reinforce its optimization tendencies toward the critical traffic flow directions during training. Despite minor performance disadvantages in certain directions, the DDQN-based control method still demonstrates stronger overall global optimization and robustness.

Table 4 presents the average travel time for vehicles under the three control strategies. The results show a high degree of consistency between the comparison of average travel time and average queue length across the different strategies. Although the DDQN-based control strategy exhibits some performance disadvantages in certain traffic directions, it achieves significant improvements in traffic efficiency across most directions. These results thoroughly validate the potential application of the proposed method in complex intersection traffic management, and demonstrate a clear advantage over traditional rule-based control strategies. It is important to note that, in high-demand traffic directions, there is still room for

Table 4. Average travel time under the three control strategies.

Traffic movement	Fixed-time control	Vehicle-actuated control	DDQN-based control
SBT-NBT	**21.63**	37.06	24.29
WBT-EBT	51.26	**29.03**	42.60
SBL-NBL	56.71	32.10	**16.36**
WBL-EBL	57.16	32.49	**17.91**

The bold value represents the minimum value, indicating that under the corresponding control method, this traffic flow direction has the best control traffic efficiency.

improvement in the DDQN-based control strategy. Future research could consider introducing traffic-sensitive reward weighting mechanisms to enhance the optimization tendency of the strategy for key traffic directions, thereby achieving more precise resource allocation and further improving traffic efficiency.

We further investigate the performance of the proposed control method under different demand levels to assess its adaptability across varying traffic scenarios. Different demand levels are generated by multiplying the baseline traffic demand by different demand coefficients. The baseline traffic demand corresponds to a low-demand scenario, a demand coefficient of 1.2~1.4 corresponds to a medium-demand scenario, and demand coefficients greater than 1.4 correspond to high-demand scenarios.

Figure 9 illustrates the average queue length at the intersection under the three control strategies across different traffic demand levels. Overall, the queue length increases for all strategies as demand intensifies, reflecting the substantial pressure that high-volume conditions place on intersection capacity. Among the three methods, the fixed-time controller is the most sensitive to traffic growth. Its average queue length rises sharply and reaches nearly 45 vehicles when the demand coefficient is 1.8, indicating a lack of adaptability to rapidly changing traffic conditions. The vehicle-actuated controller performs more effectively under low and medium demand, maintaining relatively stable queues with a slower growth trend. However, its performance deteriorates as demand becomes high, where fluctuations in detector inputs lead to less efficient phase adjustments and noticeably longer queues. In contrast, the DDQN-based control strategy demonstrates the strongest adaptability across all demand levels. Although queue lengths increase under high-volume conditions, the growth rate is significantly lower than that of the fixed-time and vehicle-actuated controllers. This indicates that the DDQN-SE controller can better allocate green time dynamically, prevent excessive queue accumulation, and maintain more stable control performance under congested traffic.

It is important to note that while the reinforcement learning-based signal control method effectively reduces the average queue length at the intersection overall, its performance has not yet reached optimal levels for certain traffic directions. Figure 10 shows the variation in average queue length at the WBT-EBT direction under the three control strategies across different traffic demand levels. It can be observed that as the demand coefficient increases, the queue lengths under all three strategies show an upward trend. Among them, fixed-time control exhibits the most rapid increase in queue length, with the overall performance being the worst. At high demand levels, it leads to severe congestion and queue buildup, demonstrating its lack of ability to adjust to changes in traffic flow. In contrast, actuated signal control maintains lower queue lengths across all demand levels, with particularly good performance at demand coefficients of 1.6 and 1.8, which reflects its capacity to dynamically adjust signal timings in response to fluctuating traffic volumes. This ability to recognize traffic flow variations and adapt signal timings in real time gives actuated control a strong advantage in dynamic traffic environments. The DDQN-based strategy performs relatively stably under medium- and low-demand conditions, with queue lengths similar to actuated control and clearly outperforming fixed-time control. However, at high-demand levels, the queue length increases slightly more than in vehicle-actuated control, indicating some performance limitations. This phenomenon highlights that the current RL model still has room for improvement in handling high-demand traffic directions.

Computational complexity and real-time performance

To evaluate the deployability of the proposed DDQN-SE controller in real-world traffic signal systems, the computational complexity and operational efficiency were further analyzed. The final trained model contains approximately 1.2 million parameters, which corresponds to a lightweight neural network architecture compared with typical deep reinforcement learning models.

Training was conducted on an NVIDIA RTX 3090 GPU, and 1,500 episodes required approximately 4.8 h, demonstrating that the model can be trained efficiently offline. More importantly, the inference time during deployment—when the agent selects an action based on the current state—is extremely low. The average inference time per decision is 3.1 ms, which is significantly below the typical signal controller update interval (≥ 1 s). This ensures that the DDQN-SE controller can operate comfortably within real-time constraints.

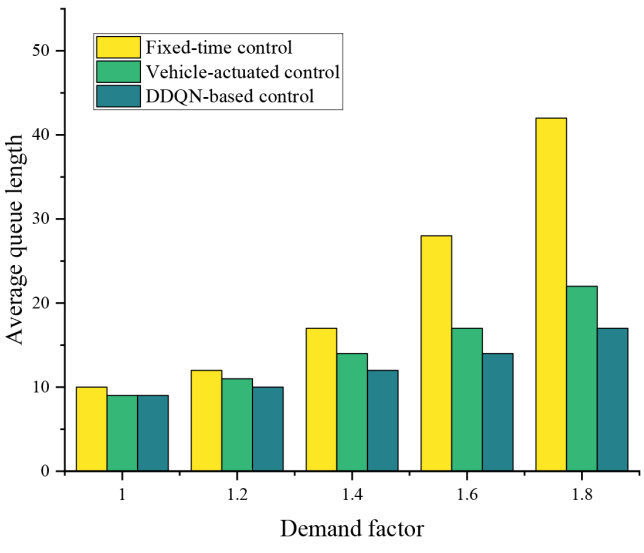

Fig. 9 Average queue length at the intersection under different demand levels.

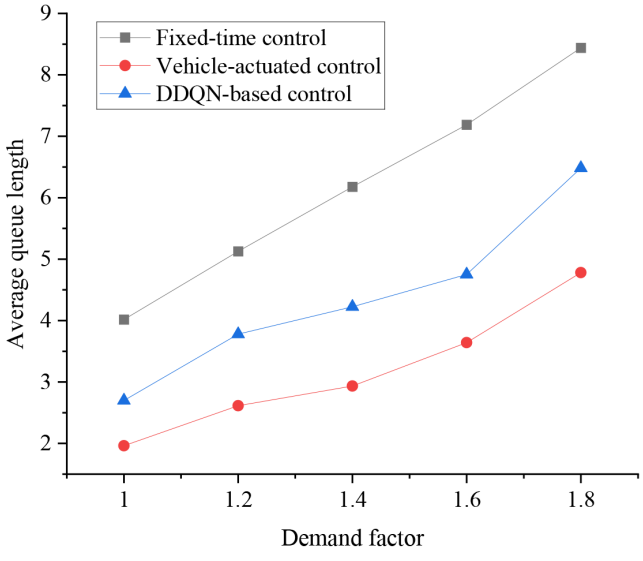

Fig. 10 Average queue length in the WBT-EBT direction under different demand levels.

Overall, the results indicate that the proposed method offers a good balance between control performance and computational efficiency. The model is lightweight enough for real-time deployment and can be retrained periodically as traffic conditions evolve, making it suitable for practical intelligent signal control applications.

Conclusions

This paper addresses the issues of weak responsiveness and rigid control strategies in urban intersection signal control under complex traffic environments. A Double Deep Q-Network (DDQN)-based reinforcement learning method is proposed, achieving the joint optimization of signal phase sequence and base duration. The method enhances the agent's perception accuracy and learning efficiency by constructing a state space that integrates key traffic indicators, such as queue length, average speed, and waiting time, and by introducing the principle of consistency between state and reward definitions. Simulation results show that the proposed method significantly reduces average queue length and vehicle travel time in most traffic directions compared to fixed-time control and vehicle-actuated control methods. Sensitivity analysis under different traffic demand levels reveals the importance of appropriately setting the base phase duration to ensure control performance. Robustness evaluation further confirms the stability of the proposed method in high-demand scenarios.

Future research can focus on the following two aspects: first, designing a dynamically weighted reward function based on traffic pressure, incorporating traffic flow distribution characteristics to improve the agent's responsiveness to key traffic directions; and second, extending the method to multi-intersection cooperative control scenarios, further exploring the potential of deep reinforcement learning in complex traffic systems.

Author contributions

The authors confirm contributions to the paper as follows: study conception and design: Dai R; analysis and interpretation of results: Li Y, Dai R; draft manuscript preparation: Dai R, Li R. All authors reviewed the results and approved the final version of the manuscript.

Acknowledgments

This research was supported by National Natural Science Foundation of China (Grant No. 52402373), Shandong Provincial Natural Science Foundation (Grant No. ZR2024QG016), Fund of National Engineering Research Center for Water Transport Safety (No. A202502), and the China Postdoctoral Science Foundation under (Grant No. BX20230203).

Conflict of interest

The authors declare that they have no conflict of interest.

Dates

Received 12 August 2025; Revised 25 November 2025; Accepted 11 February 2026; Published online 31 March 2026

References

[1] Wu J, Qu X. 2022. Intersection control with connected and automated vehicles: a review. *Journal of Intelligent and Connected Vehicles* 5:260–269

[2] Florin R, Olariu S. 2015. A survey of vehicular communications for traffic signal optimization. *Vehicular Communications* 2:70–79

[3] Hamilton A, Waterson B, Cherrett T, Robinson A, Snell I. 2013. The evolution of urban traffic control: changing policy and technology. *Transportation Planning and Technology* 36:24–43

[4] Webster FV. 1958. *Traffic signal settings*. London, England: Her Majesty's Stationery Office

[5] Wong CK, Wong SC. 2003. Lane-based optimization of signal timings for isolated junctions. *Transportation Research Part B: Methodological* 37:63–84

[6] Yin Y. 2008. Robust optimal traffic signal timing. *Transportation Research Part B: Methodological* 42:911–924

[7] Yun I, Park BB. 2012. Stochastic optimization for coordinated actuated traffic signal systems. *Journal of Transportation Engineering* 138:819–829

[8] Dai R, Cai P, Wang X, Zhang R. 2023. A computationally efficient and refined signal control method for isolated intersections in a connected vehicle environment. *Expert Systems with Applications* 234:121073

[9] Mercader P, Uwayid W, Haddad J. 2020. Max-pressure traffic controller based on travel times: an experimental analysis. *Transportation Research Part C: Emerging Technologies* 110:275–290

[10] Abed Al Raheem Magableh A, Almakhadmeh MA, Alsrehin N, Klaib AF. 2020. Smart traffic light management systems: a systematic literature review. *Journal International Journal of Technology Diffusion* 11:22–47

[11] Touhbi S, Babram MA, Nguyen-Huu T, Marilleau N, Hbid ML, et al. 2017. Adaptive traffic signal control: exploring reward definition for reinforcement learning. *Procedia Computer Science* 109:513–520

[12] Joo H, Ahmed SH, Lim Y. 2020. Traffic signal control for smart cities using reinforcement learning. *Computer Communications* 154:324–330

[13] Acar B, Sterling M. 2023. Ensuring federated learning reliability for infrastructure-enhanced autonomous driving. *Journal of Intelligent and Connected Vehicles* 6:125–135

[14] Sun H, Chen C, Liu Q, Zhao J. 2020. Traffic signal control method based on deep reinforement learning. *Computer Science* 47:169–174

[15] La P , Bhatnagar S. 2011. Reinforcement learning with function approximation for traffic signal control. *IEEE Transactions on Intelligent Transportation Systems* 12:412–421

[16] Shabestary SMA, Abdulhai B. 2022. Adaptive traffic signal control with deep reinforcement learning and high dimensional sensory inputs: case study and comprehensive sensitivity analyses. *IEEE Transactions on Intelligent Transportation Systems* 23:20021–20035

[17] Haddad TA, Hedjazi D, Aouag S. 2022. A deep reinforcement learning-based cooperative approach for multi-intersection traffic signal control. *Engineering Applications of Artificial Intelligence* 114:105019

[18] Shabestary SMA, Abdulhai B. 2018. Deep learning vs. discrete reinforcement learning for adaptive traffic signal control. *Proc. 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 2018*. US: IEEE. pp. 286–293 doi: 10.1109/ITSC.2018.8569549

[19] Liang X, Du X, Wang G, Han Z. 2019. A deep reinforcement learning network for traffic light cycle control. *IEEE Transactions on Vehicular Technology* 68:1243–1253

[20] Nigam A, Srivastava S. 2023. Hybrid deep learning models for traffic stream variables prediction during rainfall. *Multimodal Transportation* 2:100052

[21] Bouktif S, Cheniki A, Ouni A, El-Sayed H. 2023. Deep reinforcement learning for traffic signal control with consistent state and reward design approach. *Knowledge-Based Systems* 267:110440

[22] Liang X, Guler SI, Gayah VV. 2020. An equitable traffic signal control scheme at isolated signalized intersections using Connected Vehicle technology. *Transportation Research Part C: Emerging Technologies* 110:81–97

[23] Chu T, Wang J, Codecà L, Li Z. 2020. Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE Transactions on Intelligent Transportation Systems* 21:1086–1095

[24] Aslani M, Mesgari MS, Wiering M. 2017. Adaptive traffic signal control with actor-critic methods in a real-world traffic network with different traffic disruption events. *Transportation Research Part C: Emerging Technologies* 85:732–52

[25] Haydari A, Yilmaz Y. 2022. Deep reinforcement learning for intelligent transportation systems: a survey. *IEEE Transactions on Intelligent Transportation Systems* 23(1):11–32

[26] El-Tantawy S, Abdulhai B. 2010. An agent-based learning towards decentralized and coordinated traffic signal control. *Proc. 13th International IEEE Conference on Intelligent Transportation Systems, Funchal, Portugal, 2010*. US: IEEE. pp. 665–670 doi: 10.1109/ITSC.2010.5625066

[27] Mousavi SS, Schukat M, Howley E. 2017. Traffic light control using deep policy-gradient and value-function-based reinforcement learning. *IET Intelligent Transport Systems* 11:417–423

[28] Khamis MA, Gomaa W. 2013. Enhanced multiagent multi-objective reinforcement learning for urban traffic light control. *Proc. 2012 11th International Conference on Machine Learning and Applications, Boca Raton, FL, USA, 2012*. US: IEEE. pp. 586–591 doi: 10.1109/ICMLA.2012.108

[29] Noaeen M, Naik A, Goodman L, Crebo J, Abrar T, et al. 2022. Reinforcement learning in urban network traffic signal control: a systematic literature review. *Expert Systems with Applications* 199:116830

[30] Van Hasselt H, Guez A, Silver D. 2016. Deep reinforcement learning with double Q-learning. *Proceedings of the AAAI Conference on Artificial Intelligence* 30(1):2094–2100

[31] Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518:529–533