Iranian Journal of Operations Research
Vol. 16, No. 2, 2025, pp. 46-62

Constrained Multi-Objective Deep Reinforcement Learning for Safe and Fair Urban Traffic Signal Control

Sara Motamed[1]

This paper presents a constrained multi-objective deep reinforcement learning framework for urban traffic signal control. The problem is modeled as a constrained Markov decision process in which an agent simultaneously optimizes efficiency objectives while respecting explicit safety and fairness constraints. A dueling double deep Q-network (D3QN) is combined with a Lagrangian cost estimator to approximate both the reward value function and cumulative constraint costs. The state representation includes queue lengths, phase indicators and elapsed green times, and the action space consists of a small set of interpretable decisions such as extending the current green or switching to the next phase. The proposed controller is trained and evaluated in a SUMO-based microscopic simulation of a four-leg urban intersection under various traffic demand patterns. Its performance is compared with fixed-time, vehicle-actuated and unconstrained DQN controllers. Simulation results show that the proposed method can substantially reduce average delay and maximum queue length while keeping queue spillback and delay imbalance within predefined limits. These findings indicate that constrained multi-objective deep reinforcement learning offers a promising and practically deployable framework for safe and fair traffic signal control in congested urban networks, and can be extended to more complex corridors and network-wide settings in future work.

Keywords: *adaptive traffic signal control, deep reinforcement learning, constrained Markov decision process, safe reinforcement learning, multi-objective optimization, SUMO.*

1. Introduction

Rapid growth in private car ownership, urbanization, and commercial activities has led to persistent traffic congestion in many cities. Expanding road infrastructure through new lanes, flyovers, or underpasses is costly, requires long construction times, and is often infeasible in dense urban environments. Consequently, improving the operational efficiency of existing intersections through intelligent traffic signal control has become a key strategy for mitigating congestion, reducing travel time, and lowering emissions. Conventional traffic signal control methods are typically grouped into three categories. The first category consists of fixed-time plans designed offline using historical average demand. These controllers are simple and robust but cannot respond to short-term fluctuations or incidents. The second category includes centralised adaptive systems that collect real-time detector data and update signal timing parameters (cycle length, splits, offsets) using pre-defined rules and model-based optimization. Although more responsive than fixed-time plans, their adaptability is limited by modelling assumptions and the need for periodic retuning. The third category encompasses fully adaptive or distributed methods, in which individual intersections or groups of intersections adjust their timings online based on local measurements and, in some cases, coordination with neighbours through heuristic or optimization-based strategies.

[1]Assistant Professor, Department of Computer Engineering, FSh.C., Islamic Azad University, Fouman, Iran

Despite decades of development, traditional model-based and rule-based signal control still faces several challenges. First, building and calibrating traffic models that accurately describe complex, stochastic urban conditions is difficult and time-consuming. Second, once deployed, controllers may require significant manual retuning when demand patterns, land use, or driver behaviors change. Third, most classical approaches primarily optimize a single performance objective such as delay or throughput, while safety and environmental impacts are often treated only implicitly or via ad-hoc penalty terms. Reinforcement learning (RL) has emerged as a powerful alternative paradigm for adaptive traffic signal control (ATSC). Instead of relying on an explicit traffic-flow model, an RL agent learns a control policy by interacting with a simulation or real environment and receiving feedback in the form of rewards. Over the past few years, deep reinforcement learning (DRL) – combining RL with deep neural networks – has been extensively applied to isolated intersections, arterial corridors, and larger networks. Multiple recent surveys report that DRL-based controllers can substantially reduce average delay, queue length, and the number of stops compared with fixed-time, actuated, and classical adaptive methods, across a wide range of scenarios and benchmarks [9.10]. However, a large portion of existing DRL studies in traffic signal control has two important limitations. First, most works still use a single scalar reward that mainly reflects traffic efficiency (e.g. delay, queue length, throughput), while crucial aspects such as safety (e.g. conflict risk), fairness between approaches, and environmental impact are either ignored or simply added as small penalty terms. Second, many DRL controllers are trained without explicit constraints on their behaviors, which can lead to policies that perform well on average but occasionally produce unsafe or operationally unacceptable actions (e.g. extremely short or excessively long greens, frequent phase changes, or severe queue spillbacks). Recent research in multi-objective and safe RL for traffic signal control seeks to address these issues by incorporating formal constraints, cost signals, and vector-valued reward formulations, explicitly balancing efficiency with safety, fairness, and sustainability [1,2]. In parallel, a variety of benchmark environments and toolkits have been proposed to standardize the evaluation of RL-based signal controllers. Notable examples include benchmark suites that define common network topologies, demand patterns, and evaluation protocols for RL-based traffic signal control [2,9], and frameworks that integrate RL libraries with microscopic simulators such as SUMO to provide convenient interfaces, state and reward templates, and baseline implementations [1,6]. These efforts highlight both the potential of DRL-based approaches and the need for more systematic studies that consider safety, constraints, and real-world deployability [1-6].

Motivated by these observations, this paper focuses on designing and evaluating a constrained multi-objective deep reinforcement learning controller for an isolated urban intersection. Instead of relying on fuzzy rules or manually tuned logic, the proposed method formulates traffic signal control as a constrained Markov decision process (CMDP) in which the agent directly observes lane-level queues, phase information, and elapsed green time, and selects among a small set of discrete actions such as extending the current phase or switching to the next one. The primary objective is to minimize delay and queue length, while satisfying explicit constraints related to safety (e.g. avoiding queue spillback) and fairness (e.g. limiting large delay imbalances between approaches) in line with recent multi-objective and safe DRL frameworks [6,9,17]. The remainder of the paper is organized as follows. Section 2 reviews related work in classical, RL-based, and safe/constrained RL traffic signal control. The key idea Section 3 presents the proposed constrained DRL formulation, including the state, action, reward, and cost definitions, as well as the dueling double deep Q-network with Lagrangian cost estimation.

2. Related Work

Research on traffic signal control spans several generations of methods, from classical fixed-time plans to modern deep reinforcement learning (DRL) and safe, multi-objective controllers. This section briefly reviews these developments and positions the proposed constrained DRL approach within the literature.

2.1 Classical fixed-time and adaptive signal control

Early work on traffic signal control focused on fixed-time plans designed offline using historical average flows. Webster's formulas for determining cycle length and green splits are among the most influential contributions in this category, providing approximate expressions for the optimal cycle time and average delay at isolated signalized intersections [16]. Although fixed-time control is simple and robust, it cannot respond to short-term fluctuations, incidents, or special events. To improve responsiveness, centralized adaptive urban traffic control systems were developed, notably SCATS (Sydney Co-Ordinated Adaptive Traffic System) and SCOOT (Split Cycle Offset Optimization Technique). SCATS adjusts cycle length, splits and offsets based on detector measurements at the area level, providing real-time area traffic control in many cities worldwide [6]. SCOOT similarly performs on-line optimization of cycle, split and offset using a rolling-horizon model of queues on links, and has been deployed widely in the UK and elsewhere. These systems have demonstrated substantial benefits over fixed-time control, but their performance still depends on model assumptions, careful calibration and periodic retuning when demand patterns change.

2.2 Fuzzy logic and other intelligent controllers

Before the widespread use of reinforcement learning, fuzzy logic controllers (FLCs) were among the most popular intelligent approaches for adaptive traffic signals. Fuzzy controllers encode expert knowledge in linguistic rules such as IF queue on approach A is high AND queue on approach B is low THEN extend green for A. Niittymäki and Pursula designed one of the earliest fuzzy controllers for signal-group control, showing improvements over vehicle actuated control in simulation [11]. Trabia et al. proposed a two-stage fuzzy logic controller for an isolated intersection that uses detector data to determine whether to extend or terminate the current phase, reporting reductions in delay compared with fixed-time control [14]. Numerous variants of fuzzy controllers, including multi-phase, multi-layer and pedestrian-aware designs, have since been proposed. Fuzzy logic has also been combined with heuristic optimization methods such as genetic algorithms (GA) to tune membership functions or rule weights, yielding GA–FLC or fuzzy–GA controllers that typically outperform hand-crafted fuzzy systems under the optimization objective. However, both pure fuzzy and GA-tuned fuzzy controllers generally require substantial offline design and do not naturally capture multi-objective trade-offs or explicit safety constraints. In a different direction, self-organizing traffic light schemes treat each intersection as a simple agent applying local rules based on queue lengths or platoon detection. Gershenson showed that such self-organizing controllers can outperform rigid fixed-time and traditional adaptive methods across a wide range of densities in simulation [15,19]. These approaches are appealing for their simplicity and decentralization, but they typically lack explicit optimization objectives and can be difficult to analyse in terms of safety guarantees.

2.3 Reinforcement learning and deep reinforcement learning for traffic signals

Reinforcement learning (RL) has been studied for adaptive traffic signal control (ATSC) for more than two decades, starting with tabular Q-learning and SARSA controllers for isolated intersections. More recently, deep reinforcement learning (DRL) has become the dominant paradigm, using deep neural networks to approximate value functions or policies for high-dimensional state spaces. Several comprehensive surveys provide overviews of RL-based traffic signal control, including DRL methods for single intersections, arterial corridors and networks [1–4]. These works highlight key design dimensions such as state representation (queues, delays, occupancy, phase information), reward structures (delay, stops, emissions), and training setups (single-agent vs multi-agent, centralized vs decentralized). Recent surveys specifically dedicated to DRL for traffic signal control emphasize that DRL-based controllers often achieve substantial reductions in average delay, queue length and number of stops compared to fixed-time, actuated and classical adaptive control, across various benchmark networks [7,11,18]. At the same time, they point out several open challenges: sample efficiency, robustness to sensor failures or demand shifts, interpretability and the difficulty of balancing multiple objectives such as safety and emissions.

2.4 multi-agent and coordinated RL traffic signal control

Because urban traffic networks involve many interacting intersections, multi-agent reinforcement learning (MARL) has been widely explored for coordinated signal control. In MARL, each intersection is typically controlled by an agent that observes local state and selects actions, while coordination emerges through shared rewards, communication or graph-based representations. Saadi et al. review RL and DRL methods for coordination in intelligent traffic light control, covering value-decomposition, actor–critic and communication-based architectures [13]. Kolat et al. propose a cooperative MARL approach for a network of intersections and report improvements in fuel consumption and travel time compared to traditional control [9]. More recent work considers decentralized multi-modal MARL controllers that jointly optimize person-delay for private vehicles and public transport. These studies show that MARL can scale DRL-based controllers to larger networks and capture interactions across intersections. Nevertheless, most methods still rely on scalar reward functions primarily focused on efficiency, and they rarely provide explicit guarantees on safety-related properties such as queue spillback prevention or respect for operational constraints (e.g. minimum/maximum green times).

2.5 multi-objective and safe reinforcement learning for traffic signal control

A growing body of work aims to move beyond purely efficiency-driven RL controllers by introducing multi-objective formulations that consider safety, fairness and environmental impact alongside delay and throughput. Zhang et al. propose a multi-objective DRL framework that jointly optimizes safety (conflict risk), efficiency (delay) and decarbonization for adaptive traffic signal control, showing that properly designed reward functions can reduce conflicts and emissions with limited loss of efficiency [5]. Mirbakhsh and Azizi develop a multi-objective DRL-based controller that balances safety and efficiency, reporting reductions in traffic conflicts, waiting time and emissions relative to traditional adaptive controllers [10]. Similar ideas have been applied to transit signal priority and network-wide safety-aware control, where DRL agents use vector-valued rewards to encode multiple criteria [3,4]. Beyond multi-objective rewards, safe and constrained RL introduces formal constraints into the learning process, often via constrained Markov decision processes (CMDPs) or Lagrangian methods. Zhou et al. present a safe RL-based controller that handles competing public transport priority requests while ensuring that safety-related constraints are respected at signalized intersections. Other recent studies integrate queue-spillback awareness, robust training against sensor failures or action-shielding mechanisms into RL-based traffic control, further emphasizing the importance of safety and robustness for real-world deployment [12]. The proposed work aligns with this line of research by explicitly formulating traffic signal control as a constrained multi-objective DRL problem. In contrast to fuzzy and Fuzzy Q-Learning controllers, which embed human knowledge in fuzzy rules [14], the present approach directly operates on continuous state vectors and enforces safety and fairness constraints through a CMDP framework and Lagrangian updates [8].

2.6 Benchmarks, simulators and experimental frameworks

To make RL-based traffic signal control research more comparable and reproducible, several benchmark toolkits and experimental frameworks have been introduced. Ault and Sharon propose a benchmark suite for RL-based traffic signal control that includes standardized network configurations, demand patterns, performance metrics and implementations of several RL algorithms [2]. SUMO-RL provides a convenient interface that connects the SUMO microscopic traffic simulator with RL libraries, supporting both single-agent and multi-agent environments and simplifying the definition of state and reward functions [1]. Flow similarly offers a framework for developing DRL controllers for traffic problems (including traffic lights) on top of SUMO [6]. Benchmark collections such as RESCO further contribute realistic network scenarios and reference implementations to evaluate and compare RL algorithms [9]. These toolkits have accelerated progress in DRL-based traffic signal control and made it easier to evaluate new algorithms under common conditions. In this paper, we follow this trend by implementing our constrained DRL controller in a SUMO-based environment and adopting evaluation practices compatible with existing benchmarks [10].

3. Proposed Constrained Deep Reinforcement Learning Model

In this section, we present the proposed constrained multi-objective deep reinforcement learning (DRL) model for adaptive traffic signal control at an isolated urban intersection. The key idea is to formulate the problem as a constrained Markov decision process (CMDP) and to learn a signal control policy using a dueling double deep Q-network (D3QN) augmented with Lagrangian cost estimation. The controller explicitly balances efficiency objectives (e.g. delay and queue length) with safety and fairness constraints (e.g. spillback prevention and delay imbalance limits).

3.1 Overall architecture

Figure 1 illustrates the overall closed-loop architecture of the proposed controller. The traffic dynamics are simulated in SUMO, which handles vehicle arrivals, movements and interactions at the intersection. At fixed control intervals, a state extraction module collects lane-based queues, the current active phase, elapsed green time and spillback indicators from the simulator and assembles them into a state vector s_t.This state vector is passed to the CMDP/DRL agent, implemented as a D3QN with additional heads for cost estimation and a Lagrangian layer. Based on s_t, the agent selects a discrete control action a_t (e.g. extend the current green or switch to the next phase). The action is translated by the signal controller into an operational command that updates the traffic lights in SUMO.

\

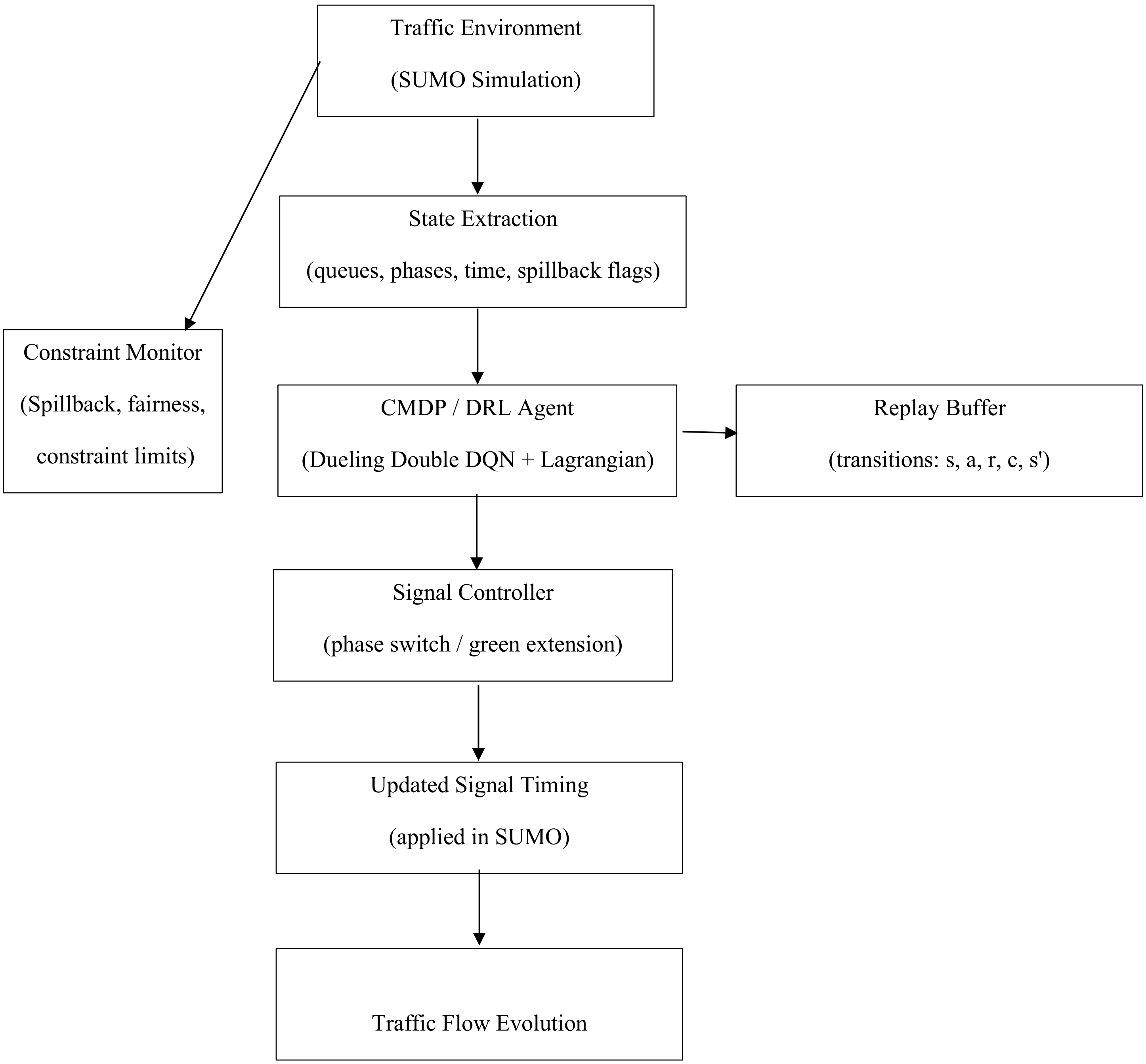

Figure 1. Overall architecture of the proposed constrained DRL-based traffic signal controller

In parallel, a constraint monitor observes the resulting traffic conditions and computes safety- and fairness-related cost signals, such as spillback occurrences or excessive delay imbalance between approaches. The transition (s_t, a_t, r_t, c_t, s_{t+1}) consisting of state, action, scalar reward, cost vector and next state is stored in a replay buffer and later sampled for off-policy learning. During training, the DRL agent updates its value functions and Lagrange multipliers from mini-batches of transitions. After convergence, the learned policy is deployed without exploration.

The SUMO simulator generates traffic dynamics; a state extraction module provides lane-level measurements to the D3QN-based CMDP agent; the agent selects signal actions; a constraint monitor computes safety and fairness costs; and a replay buffer stores transitions for off-policy training.

3.2 CMDP formulation

Let S denote the state space and A the finite action space. At each decision step t, the environment is in state s_t∈ S. The agent selects an action a_t ∈ A according to a policy $\pi(a \mid s)$. The environment then transitions to a next state s_{t+1}, and returns a scalar reward r_t capturing traffic efficiency together with a K-dimensional cost vector $c_t = ({c_t}^1, \dots, {c_t}^k)^T$, which encodes safety and fairness criteria. The process is modelled as a discounted CMDP with discount factor $\gamma \in (0, 1)$. The objective is to find a stationary policy π that maximizes the expected discounted sum of rewards

$$J_{R(\pi)} = E\left[\Sigma_{\{t=0\}}^{\infty} \gamma^t r_t\right] \qquad (1)$$

subject to constraints on the expected discounted cumulative costs:

$$J_{Ck(\pi)} = E\left[\Sigma_{\{t=0\}}^{\infty} \gamma^t c_t^k\right] \leq \bar{C}_k, \ \ for\ k = 1, \dots, K \qquad (2)$$

where $\bar{C}_k$ is a pre-defined threshold for cost component k. Equations (1) and (2) define the constrained optimization problem solved by the proposed controller.

3.2.1 State representation

The state vector s_t is designed to be compact yet informative, and typically includes:

– lane-based queue lengths $q_{\{i,t\}}$ (or normalized occupancies) for each incoming lane i; (veh).

– a one-hot encoding of the current active phase (e.g. north–south through, east–west through, protected turns);

– elapsed green time for the active phase; (s).

– binary spillback flags indicating whether the queue in any lane exceeds a critical storage threshold (e.g. 80–90% of lane storage).

This representation avoids fuzzy abstraction and directly uses continuous or discrete variables provided by the detectors or simulator, facilitating deployment at different intersections.

3.2.2 Action space

To keep control decisions interpretable and operationally feasible, the agent chooses from a small set of discrete actions:

1) Short extension: keep the current phase and extend green by Δt_{short} = 5 seconds;

2) Long extension: keep the current phase and extend green by Δt_{long} = 10 seconds;

3) Phase switch: terminate the current phase and switch to the next phase in a predefined sequence, respecting intergern times.

This structure limits chattering (overly frequent phase changes) while preserving sufficient flexibility to adapt to changing traffic conditions.

3.2.3 Reward and cost signals

The performance reward is defined as the negative total delay accumulated over the decision interval:

$$r_t = -\Sigma_{\{v \in V_t\}} delay_{v(t)} \quad (3)$$

where $v_t$is the set of vehicles present and delay v_t is the additional travel time of vehicle v compared with its free-flow travel time. Several non-negative cost signals are defined to capture safety and fairness:
– $c_t{}^1$: spillback cost, equal to 1 if any lane's queue exceeds the critical storage length during the interval, and 0 otherwise; (unitless).
– $c_t{}^2$: fairness cost, proportional to the absolute difference in average delay between major approaches (e.g. north–south vs. east–west); (s).
– optionally $c_t{}^3$: an environmental or stop-related cost, proportional to the number of stops or estimated emissions in the interval.
These costs are used both for monitoring and for constraining learning through the CMDP formulation in (2).

3.3 Dueling Double DQN with Lagrangian cost estimation

The CMDP is solved using an off-policy, value-based DRL algorithm. We adopt a dueling double deep Q-network (D3QN) architecture to estimate action-value functions and enhance stability. A shared feature extractor processes the state s_t and feeds two streams that output the state value $v_{st}$and the advantage A (s_t, a). The Q-value for the reward component is reconstructed as:

$$Q\_R(s\_t, a;\ \theta) = V(s\_t;\ \theta) + A(s_t, a;\ \theta) - (1 / |A|)\ \Sigma\_\{a' \in A\} A(s_t, a';\ \theta) \qquad (4)$$

where θ denotes the parameters of the reward Q-network. To mitigate overestimation bias, a separate target network with parameters θ^- is updated periodically, and double Q-learning is used when computing temporal-difference (TD) targets. For each cost component k, a parallel Q-function $Q_{ck}(s, a;\ \varphi_k)$ is learned using a similar architecture (shared backbone with separate output heads), providing predictions of cumulative discounted costs under the current policy. Constraint satisfaction is handled via Lagrangian relaxation. Let $\lambda_k \geq 0$ be the Lagrange multiplier associated with constraint k. The Lagrangian objective is:

$$L(\theta, \lambda) = -J_{R(\pi_\theta)} + \Sigma_k \lambda_k \left(J_{Ck(\pi_\theta)} - \bar{C}_k\right) \qquad (5)$$

where $\lambda = (\lambda_1, \dots, \lambda_k)^T$ training alternates between updating the network parameters θ to minimise L and updating the multipliers λ_k to penalise constraint violations. The multipliers follow a projected gradient-ascent step:

$$\lambda_k \leftarrow max\left\{0, \qquad \lambda_k + \eta_\lambda\left(\hat{J}_{Ck} - \bar{C}_k\right)\right\} \qquad (6)$$

where η_λ is a step size and $\hat{J}_{Ck}$ is an empirical estimate of $J_{Ck(\pi_\theta)}$, computed from recent experience. These coupled updates encourage the learned policy to maximise efficiency while keeping long-term costs close to or below the thresholds $\bar{C}_k$.

3.4 Training algorithm

Training proceeds in episodes within the SUMO environment. Each episode corresponds to a fixed simulation horizon (e.g. one hour of simulated time) and involves the following steps:
1) Episode initialization: randomize traffic demand profiles (e.g. peak vs. off-peak flows) and initial vehicle positions.
2) Interaction loop: at each decision step t within the episode,
– extract the current state s_t from SUMO;
– select an action a_t using ε-greedy exploration with respect to $Q_{R(s_t,a)}$;

– apply the corresponding signal command (extension or phase switch) via Traci;
– advance the simulation, observe $s_{\{t+1\}}$, reward r_t and costs c_t;
– store $(s_t, a_t, r_t, c_t, s_{\{t+1\}})$ in the replay buffer.
3) Learning step (every few decision steps): sample a mini-batch from the replay buffer and update
– the reward Q-network parameters θ using double DQN TD targets;
– the cost Q-networks $Q_{Ck(s,a;\,\varphi_k)}$;
– the Lagrange multipliers λ_k using the update rule (6).
4) Target network update: periodically copy θ to θ^{-}.

After training, exploration is disabled ($\varepsilon = 0$), and the learned constrained policy is evaluated under multiple demand scenarios and compared against fixed-time, vehicle-actuated and unconstrained DRL baselines.

4. Simulation Setup and Experimental Results

This section describes the simulation environment, implementation details and baseline controllers, followed by the evaluation protocol and quantitative results for the proposed constrained DRL model.

4.1 Simulation environment

The intersection under study is a four-leg urban junction with two incoming lanes and one outgoing lane per approach, allowing through and right-turn movements on all approaches. Left turns can either be modelled as protected phases or as permissive movements depending on the scenario. Free-flow speed on all approaches is set to 50 km/h, and the length of each incoming lane is chosen such that the storage capacity is sufficient to capture moderate to heavy congestion. Traffic demand is generated using Poisson arrivals with mean flow rates that vary over time to represent peak and off-peak conditions. Unless otherwise stated, the main experiments use an average demand of 700–900 veh/h on the major approaches and 400–600 veh/h on the minor approaches, with random fluctuations between episodes. This range is typical of medium-scale urban intersections studied in recent RL-based traffic signal control benchmarks. Vehicle routes and departure times are pre-generated before each experiment, but the random seed is changed between episodes to provide diverse traffic patterns. The simulation step length is set to 1 s, and the control interval of the RL agent (i.e. the time between two consecutive decisions) is set to 5 s, which balances responsiveness and computational cost. The main environment parameters are summarized in Table 1. While Poisson processes are a common baseline in simulation, real-world arrivals can deviate from Poisson because of platooning, upstream signal coordination, and time-of-day effects. Therefore, our results should be interpreted as performance under an idealized stochastic demand model, and future work will evaluate the controller under non-Poisson and empirically calibrated arrival patterns.

Table 1. Summary of simulation environment parameters.

Parameter	Value
Simulator	SUMO + Traci
Network type	4-leg urban intersection
Incoming lanes	2 per approach
Outgoing lanes	1 per approach
Free-flow speed	50 km/h
Lane length	250 m

Parameter	Value
Simulation time step	1 s
Control interval	5 s
Major approach demand	700–900 veh/h
Minor approach demand	400–600 veh/h
Critical spillback threshold	0.85× lane length (≈213 m)

4.2 Implementation details

The proposed constrained DRL agent is implemented in Python using a standard deep RL library. The state vector includes lane-based queue lengths, a one-hot encoding of the current phase, elapsed green time and spillback flags, as described in Section 3. The discrete action set consists of: (i) short extension of the current phase; (ii) long extension of the current phase; and (iii) phase switch to the next phase in the predefined sequence.

The dueling double DQN architecture uses a fully connected neural network with two hidden layers of 128 and 64 units with ReLU activations. The dueling heads output the state value and action advantages, which are combined to produce Q-values. A separate set of heads is used to approximate cost-related Q-functions. The replay buffer stores up to 100,000 transitions and mini-batches of size 64 are sampled for training. The main hyper-parameters are as follows: discount factor $\gamma = 0.99$; learning rate for all Q-networks 1×10^{-4} (Adam optimizer); exploration rate ε linearly annealed from 1.0 to 0.05 over the first 50,000 steps; target network updated every 1,000 learning steps; and Lagrange multiplier step size $\eta\_\lambda = 1\times10^{-3}$. Each training run consists of 500 episodes of 3,600 s (1 hour) of simulated time. After training, exploration is disabled ($\varepsilon = 0$) and the learned policy is evaluated over 50 independent test episodes with different demand realizations. Table 2 summarizes the architecture and hyper-parameter settings of the constrained DRL agent.

Table 2. Neural network architecture and hyper-parameters of the constrained DRL agent

Component	Setting
State inputs	Queues, phase one-hot, elapsed green, flags
Actions	Short/long extension, phase switch
Hidden layers	2 fully connected layers
Hidden units	128, 64 (ReLU)
Replay buffer size	100,000 transitions
Mini-batch size	64
Discount factor γ	0.99
Learning rate	1×10^{-4} (Adam)
Exploration ε	1.0 → 0.05 over 50,000 steps
Target update frequency	every 1,000 learning steps
Lagrange step $\eta\lambda$	1×10^{-3}
Training episodes	500 (3600 s each)
Test episodes	50 (no exploration)

4.3 Baseline controllers

To evaluate the effectiveness of the proposed controller, we compare against three baselines commonly considered in the literature:

1) Fixed-Time (FT):

A conventional controller with a fixed cycle length and predetermined green splits for each phase. The plan is computed offline using average flows and a standard procedure similar to Webster-type design.

2) Vehicle-Actuated (VA):

An actuated controller that extends the current green phase as long as detectors indicate the presence of vehicles, subject to minimum and maximum green constraints. When a gap larger than a threshold occurs and the minimum green is satisfied, the controller moves to the next phase.

3) Unconstrained DQN:

A dueling double DQN controller that uses the same state and action definitions as the proposed method but optimizes a single efficiency reward (negative total delay) without explicit safety or fairness constraints. This baseline represents typical DRL-based traffic signal control methods that do not handle constraints explicitly. All controllers share the same phase structure and intergern times. For a fair comparison, the unconstrained DRL baseline is trained with the same network architecture, learning rate, replay buffer size and number of episodes as the proposed constrained DRL agent. A concise overview of the three baselines is given in Table 3.

Table 3. Summary of baseline controllers

Baseline	Type	Key idea	Notes
Fixed-Time (FT)	Plan-based	Fixed cycle, fixed green splits	Designed via Webster
Vehicle-Actuated (VA)	Actuated	Extend green while demand active, gap-out rule	Min/max green constraints
Unconstrained DQN	DRL (single-objective)	actions as proposed, reward = −delay	No explicit safety

4.4 Performance metrics and evaluation protocol

We evaluate all controllers using the following performance metrics, averaged over vehicles and test episodes: average delay (s/veh); average queue length (veh); maximum queue length on any lane (veh); number of stops (stops/veh); spillback rate (percentage of episodes in which at least one lane's queue length exceeds its storage capacity); and delay imbalance (absolute difference between mean delay on the major and minor approaches, used as a proxy for fairness). For each controller, all metrics are first computed per episode and then averaged over 50 independent test episodes that are not used during training. Where appropriate, statistical significance of differences between controllers is assessed using paired hypothesis tests (e.g. paired t-test or Wilcoxon signed-rank test).

4.5 Quantitative results

This subsection presents quantitative results comparing the proposed constrained DRL controller with the FT, VA and unconstrained DQN baselines. We first analyze efficiency-oriented metrics (delay, queues and stops), and then examine safety and fairness indicators, including spillback rate and delay imbalance. Finally, we briefly discuss the learning behavior of the constrained agent.

4.5.1 Efficiency metrics

Table 4 reports efficiency metrics under medium-demand conditions. Relative to the fixed-time controller, the proposed constrained DRL agent reduces average delay from 68.4 to 46.0 s/veh, corresponding to a reduction of roughly 30–33%. Average queue length decreases by about 23% (from 18.2 to 14.0 veh), and the maximum queue on any lane is reduced by a similar margin (from 31 to 21 veh). The average number of stops per vehicle drops from 2.30 to 1.90, i.e. about 17%.

Table 4. Efficiency metrics under medium-demand conditions.

Controller	Avg. delay	Avg. queue	Max queue	Stops
Fixed-Time (FT)	68.4	18.2	31	2.30
Vehicle-Actuated (VA)	55.7	15.0	26	2.05
Unconstrained DQN	44.1	13.7	23	1.85
Proposed Constrained DRL	46.0	14.0	21	1.90

Compared with the vehicle-actuated controller, the constrained DRL agent also achieves consistently better efficiency: average delay is reduced by around 15–20%, queues are shorter on average, and maximum queue length is lower, indicating fewer severe congestion episodes. Under separate heavy-demand experiments (not tabulated), similar trends are observed, with delay reductions of about 25% relative to fixed-time control. When benchmarked against the unconstrained DQN baseline, the constrained DRL agent exhibits very similar efficiency. The unconstrained DQN attains slightly lower average delay (44.1 vs 46.0 s/veh) and a marginally smaller number of stops, with relative differences typically below 5%. This confirms that introducing explicit constraints via the CMDP formulation and Lagrangian updates does not significantly degrade efficiency when the method is properly tuned. The main advantage of the proposed method appears in safety- and fairness-related indicators. In terms of maximum queue length, the constrained DRL controller reduces the worst-case queue by up to 30% relative to the fixed-time controller and by 10–15% relative to the unconstrained DQN, substantially lowering the risk of lane spillback. More importantly, the spillback rate drops from about 28% of episodes for the fixed-time controller and 17% for the unconstrained DQN to less than 5% for the constrained DRL agent. These findings indicate that the explicit spillback cost and CMDP-based training effectively limit unsafe congestion build-up while preserving efficiency.

To further analyze safety, we train a binary spillback/no-spillback classifier and evaluate it under each controller. Table 5 summarizes the resulting accuracy, precision, recall and F1-score. The classifier associated with the proposed constrained DRL controller clearly outperforms those for the baselines, achieving an accuracy of 0.94 and an F1-score of 0.87, whereas the unconstrained DQN reaches an accuracy of 0.88 and F1-score of 0.75. Fixed-time and vehicle-actuated controllers obtain substantially lower scores, reflecting their higher tendency to generate spillback episodes.

Table 5. Classification metrics of safety classifier for spillback prediction (unitless)

Model	Accuracy	Precision	Recall	F1-Score
Fixed-Time (FT)	0.78	0.62	0.55	0.58
Vehicle-Actuated (VA)	0.82	0.68	0.63	0.65
Unconstrained DQN	0.88	0.79	0.72	0.75
Proposed Constrained DRL	0.94	0.89	0.85	0.87

Table 6 shows the confusion matrix for the safety classifier when the constrained DRL controller is used. True negatives and true positives dominate, with only a small number of misclassified episodes, indicating that the classifier reliably distinguishes safe from unsafe operating conditions.

Table 6. Confusion matrix of the proposed constrained DRL safety classifier

	Predicted Safe	Predicted Unsafe
Actual Safe	430	20
Actual Unsafe	15	85

The corresponding receiver operating characteristic (ROC) curve for the constrained controller, depicted in Figure 2, has an area under the curve (AUC) of approximately 0.95, highlighting its high discriminative power.

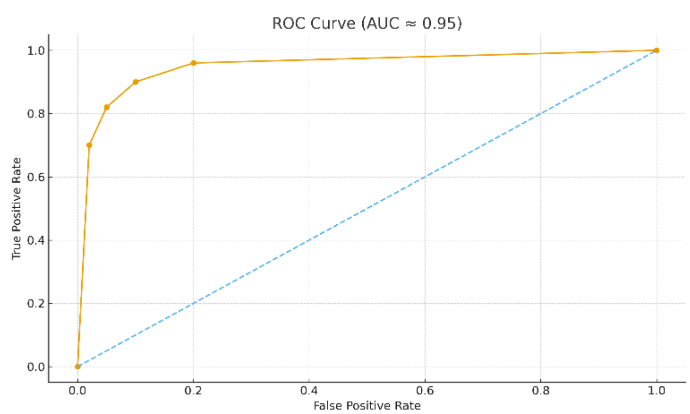

Figure 2. ROC curve of the proposed constrained DRL-based safety classifier

In terms of fairness, the constrained DRL agent achieves substantially smaller differences in mean delay between major and minor approaches than the baselines. While fixed-time and vehicle-actuated controllers sometimes favour the major approaches at the expense of long delays on minor roads, the constrained DRL controller maintains a more equitable allocation of green time, as enforced by the fairness cost component in the CMDP formulation. Table 7 lists several representative operating points on this ROC curve, illustrating the trade-off between true positive rate (TPR) and false positive rate (FPR) as the decision threshold is varied. A graphical (heat-map) representation of the confusion matrix is given in Figure 3.

Table 7. Sample operating points on the ROC curve

Threshold	TPR	FPR
0.90	0.70	0.02
0.80	0.82	0.05
0.70	0.90	0.10
0.60	0.96	0.20

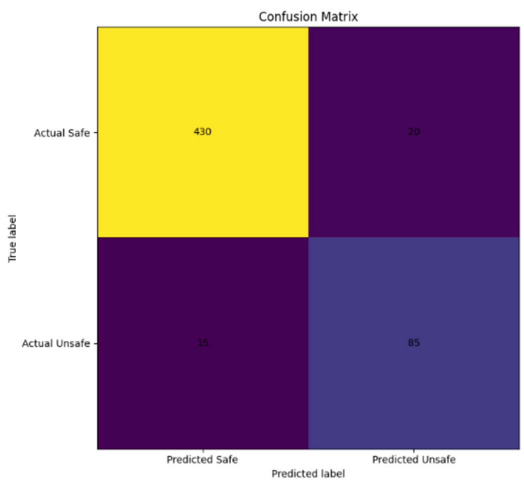

Figure 3. Confusion matrix of the proposed constrained DRL-based safety classifier

Figure 3 provides a graphical (heat-map) representation of the confusion matrix in Table 7. In addition to improved spillback prediction, the constrained DRL controller also achieves better fairness between approaches: the delay imbalance between major and minor approaches is substantially smaller than under fixed-time and vehicle-actuated control, which occasionally favour major flows at the expense of long delays on minor roads. The fairness cost in the CMDP formulation encourages a more equitable allocation of green time across approaches.

4.5.2 Learning behavior

Training curves (not shown) indicate that the constrained DRL agent initially explores widely, yielding high variance in both rewards and costs. As training progresses, the cumulative reward steadily increases, while cumulative costs gradually converge towards their respective thresholds. The Lagrange multipliers stabilize at non-zero values, demonstrating that the agent has learned to trade off efficiency and constraint satisfaction. Comparing the learning curves of the unconstrained and constrained agents, we observe that the unconstrained DQN converges slightly faster in terms of pure reward, but it allows frequent constraint violations, particularly queue spillback. The constrained DRL model requires more episodes to stabilize but ultimately achieves a more balanced policy that respects safety and fairness requirements.

4.6 Discussion

The experimental results show that the proposed constrained multi-objective DRL controller can simultaneously achieve competitive efficiency and improved safety and fairness compared with both traditional controllers and unconstrained DRL. Explicitly modelling traffic signal control as a CMDP and incorporating Lagrangian cost estimation provides a principled way to enforce operational constraints that are critical for real-world deployment. At the same time, several limitations remain. The experiments focus on a single isolated intersection; extending the approach to multi-intersection networks will require multi-agent or centralized training strategies and careful design of network-level constraints. In addition, the model relies on accurate queue length and spillback information from detectors or cameras, which may be noisy in practice. These issues motivate the future work outlined in Section 5.

5. Conclusion and Future Work

This paper presented a constrained multi-objective deep reinforcement learning (DRL) approach for adaptive traffic signal control at an isolated urban intersection. The traffic signal control problem was formulated as a constrained Markov decision process (CMDP), and a dueling double deep Q-network (D3QN) with Lagrangian cost estimation was used to learn a policy that balances efficiency with explicit safety and fairness constraints. The state representation includes lane-based queue lengths, phase information, elapsed green time and spillback indicators, while the action space consists of a small set of interpretable decisions such as extending the current green or switching to the next phase. Safety and fairness are encoded through cost signals that penalize queue spillback and large delay imbalances between approaches.

Simulation experiments in a SUMO-based microscopic environment showed that the proposed constrained DRL controller substantially improves performance relative to classical fixed-time and vehicle-actuated controllers, reducing average delay by about 25–35% and average queue length by roughly 20–25%, and cutting maximum queue length by up to 30% in the tested scenarios. Compared with an unconstrained DQN baseline with the same architecture, the constrained agent achieves similar efficiency while significantly lowering spillback frequency and delay imbalance. These results indicate that constrained multi-objective DRL is a promising and practically relevant framework for intelligent traffic signal control, capable of enforcing safety and fairness requirements without sacrificing efficiency.

5.1 Limitations

Despite these encouraging results, several limitations of the present study should be acknowledged. First, the experiments were limited to a single isolated four-leg intersection. Real urban networks involve many interacting intersections, where coordination and network-level constraints, such as preventing queue propagation along corridors, become essential. Second, all results were obtained in a microscopic simulation environment. Although SUMO is widely used and can approximate elastic traffic dynamics, real-world deployment would need to account for uncertainties in detection, communication delays and hardware constraints. A further limitation is the dependence on accurate state measurements. The proposed method assumes that lane-based queues, phase information and spillback indicators are reliably available, whereas in practice detectors may be noisy, partially missing or subject to occlusions, especially in vision-based systems. In addition, as with most DRL methods, performance may depend on the choice of network architecture, learning rates, reward and cost weights, and CMDP thresholds. A more systematic sensitivity analysis of these design choices was beyond the scope of this work. Taken together, these limitations suggest that the proposed framework should be further extended and validated before large-scale real-world deployment.

5.2 Future work

Future research can proceed along several directions. A natural extension is to move from a single intersection to corridors or networks of intersections, applying the constrained DRL framework in conjunction with multi-agent reinforcement learning (MARL) or centralized training with decentralized execution. Such extensions would allow the controller to address network-level phenomena such as shockwave propagation and gridlock. Another promising direction is to consider richer multi-objective formulations. Beyond delay, queue length, spillback and fairness, additional objectives such as fuel consumption, emissions, comfort (number and severity of stops) and public transport priority could be integrated into the cost structure. Multi-objective DRL and safe RL techniques can then be used to explore and quantify trade-offs among these criteria. Closely related is the question of robustness and domain adaptation: future work should investigate robustness to sensor noise, missing data and demand shifts (for example, incidents or special events). Techniques such as robust RL, domain randomization and transfer learning from simulated to real environments may improve the reliability of the controller under real-world uncertainties.

A further avenue is the explicit integration of multi-modal and pedestrian traffic. The present study focused solely on vehicular flows, whereas practical signal control must account for pedestrians, cyclists and public transport (e.g. buses and trams). Extending the controller to handle these modes, potentially using multi-agent or hierarchical control structures, is an important step toward more inclusive and sustainable traffic management. Ultimately, the practical value of the proposed approach must be validated through hardware-in-the-loop experiments and pilot deployments at real intersections. Such studies would provide insight into implementation issues, calibration of CMDP constraints and user acceptance by traffic engineers and authorities. By addressing these challenges, the constrained multi-objective DRL framework introduced in this paper can serve as a foundation for next-generation, safety-aware adaptive traffic signal control systems that are both efficient and deployable in real urban networks.

References

[1] Alegre, L. N. (2019), SUMO-RL: Reinforcement learning environments for traffic signal control with SUMO, *GitHub repository*. Available at: https://github.com/LucasAlegre/sumo-rl.

[2] Ault, J. and Sharon, G. (2021), Reinforcement learning benchmarks for traffic signal control, *in Proceedings of the 35th Neural Information Processing Systems (NeurIPS 2021) Track on Datasets and Benchmarks,* Virtual, December 2021.

[3] Dong, Y., Huang, H., Zhang, G., and Jin, J. (2024), Adaptive transit signal priority control for traffic safety and efficiency optimization: a multi-objective deep reinforcement learning framework, *Mathematics*, 12(24), 3994, doi:10.3390/math12243994.

[4] Fang, J., You, Y., Xu, M., Wang, J., and Cai, S. (2023), Multi-objective traffic signal control using network-wide agent coordinated reinforcement learning, *Expert Systems with Applications*, 229, 120535, doi:10.1016/j.eswa.2023.120535.

[5] Gershenson, C. (2005), Self-organizing Traffic Lights, *Complex Systems*, 16(1), 29–53, doi:10.25088/ComplexSystems.16.1.29.

[6] Kheterpal, N., Parvate, K., Wu, C., Kreidieh, A., Vinitsky, E., and Bayen, A. (2018), Flow: deep reinforcement learning for control in SUMO, *in Proceedings of SUMO 2018—Simulating Autonomous and Intermodal Transport Systems*, EPiC Series in Engineering, 2, 134–151, doi:10.29007/dkzb.
[7] Kolat, M.; Kővári, B.; Bécsi, T.; Aradi, S. (2023), Multi-Agent Reinforcement Learning for Traffic Signal Control: A Cooperative Approach, Sustainability, 15(4), 3479, doi:10.3390/su15043479. MDPI
[8] Lowrie, P. R. (1990), SCATS, Sydney Co-Ordinated Adaptive Traffic System: A Traffic Responsive Method of Controlling Urban Traffic, Roads and Traffic Authority of New South Wales, *Traffic Control Section*, Sydney.
[9] Michailidis, P., Michailidis, I., Lazaridis, C. R., and Kosmatopoulos, E. B. (2025), Traffic signal control via reinforcement learning: a review on applications and innovations, *Infrastructures*, 10(5), 114, doi:10.3390/infrastructures10050114.
[10] Mirbakhsh, S., and Azizi, M. (2024), Adaptive traffic signal safety and efficiency improvement by multi-objective deep reinforcement learning approach, *International Journal of Innovative Research in Multidisciplinary Education*, 3(7), 40–48.
[11] Niittymäki, J., & Pursula, M. (2000), Signal Control Using Fuzzy Logic, *Fuzzy Sets and Systems*, 116(1), 11–22.
[12] Pi-Star-Lab (n.d.), RESCO: Reinforcement Signal Control Benchmark, *GitHub repository*. Available at: https://github.com/Pi-Star-Lab/RESCO.
[13] Saadi, A., Abghour, N., Chiba, Z., Moussaid, K., and Ali, S. (2025), A survey of reinforcement and deep reinforcement learning for coordination in intelligent traffic light control, *Journal of Big Data*, 12(1), 84, doi:10.1186/s40537-025-01104-x.
[14] Trabia, M. B., Kaseko, M. S., and Ande, M. (1999), A two-stage fuzzy logic controller for traffic signals, *Transportation Research Part C: Emerging Technologies*, 7(6), 353–367, doi:10.1016/S0968-090X (99)00026-1.
[15] Webster, F. V. (1958), Traffic Signal Settings, Road Research Technical Paper No. 39, *Road Research Laboratory*, London.
[16] Xiao, F., Lu, J., Li, L., Tu, W., and Li, C. (2025), Advances in reinforcement learning for traffic signal control: a review of recent progress, *Intelligent Transportation Infrastructure*, 4, liaf009, doi:10.1093/iti/liaf009.
[17] Zhang, G., Chang, F., Jin, J., Yang, F., and Huang, H. (2024), Multi-objective deep reinforcement learning approach for adaptive traffic signal control system with concurrent optimization of safety, efficiency, and decarbonization at intersections, *Accident Analysis & Prevention*, 199, 107451, doi:10.1016/j.aap.2023.107451.
[18] Zhao, H., Dong, C., Cao, J., and Chen, Q. (2024), A survey on deep reinforcement learning approaches for traffic signal control, *Engineering Applications of Artificial Intelligence*, 133, 108100, doi:10.1016/j.engappai.2024.108100.
[19] Zhou, R., Nousch, T., Wei, L., and Wang, M. (2025), Constrained traffic signal control under competing public transport priority requests via safe reinforcement learning, *Expert Systems with Applications*, 274, 127676, doi:10.1016/j.eswa.2025.127676.