iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** II **Month of publication:** February 2026

DOI: https://doi.org/10.22214/ijraset.2026.76765

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue II Feb 2026- Available at www.ijraset.com

Adaptive Traffic Signal Timing Optimization for Urban Intersections Using Reinforcement Learning

Saber Muthanna Ahmed Qasem [1], Feng Wang [2]
School of Information Science and Engineering, University/Institution: Henan University of Technology, Henan, China

Abstract: ***Urban traffic congestion remains a pressing challenge, particularly in developing regions where infrastructure development struggles to keep pace with rapid urbanization. Fixed-time traffic signal systems, commonly deployed in countries like Yemen, lack the adaptability needed to manage dynamic and unpredictable traffic flows efficiently. This thesis addresses these limitations by proposing and evaluating reinforcement learning (RL)–based algorithms for adaptive traffic signal control, with a dual focus on improving traffic flow and reducing environmental impact. The research introduces two core innovations: a decentralized Q-learning algorithm and a deep reinforcement learning framework based on Proximal Policy Optimization with Masking (PPO-Mask). While Q-learning demonstrated substantial improvements over traditional control strategies in reducing vehicle delays and emissions, its scalability was limited in complex traffic environments. To overcome these challenges, the PPO-Mask model was developed, incorporating action masking to ensure safer decision-making and faster convergence in high-dimensional settings. Simulations conducted using the SUMO platform across both synthetic and real-world scenarios demonstrated that PPO-Mask consistently outperformed Q-learning and fixed-time baselines across all key performance metrics. This work contributes a robust, scalable, and cost-effective approach to adaptive traffic signal control that is particularly suitable for low-resource urban environments. It also provides a comparative framework and practical insights that can inform future integration of AI-driven traffic control strategies into broader urban mobility planning.***

Keywords: Adaptive Traffic Signal Timing, Deep Reinforcement Learning, Traffic Signal Optimization, Q-learning, SUMO Simulation, Multi-Intersection, Real-Time Traffic Management

I. INTRODUCTION

In recent years, Reinforcement Learning (RL) has emerged as a promising methodology to address these challenges in real-world traffic management. Unlike traditional methods that depend on static traffic flow models, RL approaches enable systems to learn optimal timing strategies by interacting directly with the environment. This learning process is based on continuous feedback, where an RL agent optimizes a policy to maximize a specific reward function, such as minimizing travel time or congestion [6, 7]. The ability of RL to operate without prior knowledge of traffic conditions offers a significant advantage, particularly in the face of unpredictable urban traffic scenarios.

However, applying RL to traffic signal timing introduces several complexities. Early RL-based approaches typically modeled traffic intersections as agents optimizing their reward by selecting actions such as green-light phase changes, based on simplified state representations (e.g., vehicle counts or queue lengths) and a predefined reward structure [[6, 7]. Despite promising results in simple settings, these methods face significant challenges when extended to more complex traffic scenarios. For instance, at an intersection with multiple traffic movements (e.g., left-turn, through, and right-turn lanes), the state space expands exponentially, increasing the computational complexity and the difficulty of learning optimal policies [8].

The state-space explosion becomes particularly evident in multi-phase intersections, where the number of possible actions increases rapidly. A typical 8-phase intersection, where vehicles can move in multiple directions and turns, requires the RL algorithm to explore a larger state-action space. In such cases, the RL agent must evaluate multiple conflicting actions (e.g., prioritizing certain phases over others), which often results in inefficient exploration and slower convergence to optimal solutions [9] . Moreover, trial-and-error learning in a real-world setting is costly, as ineffective policies may lead to congestion, exacerbating traffic delays and increasing environmental pollution [10].

To address the challenges of traffic signal timing, model-free reinforcement learning (RL) approaches, such as Fixed time (Yemen), Q-learning and PPO-Mask**,** have been applied. Unlike traditional model-based methods, which require learning an approximate model of the traffic environment, our potential algorithm directly learn optimal policies through interaction with the environment. Q-learning allows the agent to optimize traffic signal timings based on real-time traffic data, while it enhances this process by incorporating action masking for efficient phase transitions.

These algorithms focus on improving data efficiency and significantly accelerating the learning process by continuously interacting with the environment. This approach reduces the reliance on trial-and-error, optimizing the decision-making process in complex traffic signal timing tasks.

This research proposes an adaptive traffic signal timing system utilizing reinforcement learning (RL), specifically Q-learning. Simulating this system in the SUMO (Simulation of Urban MObility) platform using real-world traffic data, the results demonstrate that the proposed model outperforms traditional fixed-time systems and Q-learning, offering improved convergence rates, reduced vehicle waiting times, and lower emissions, particularly in complex intersection scenarios. By simulating this system in the SUMO (Simulation of Urban MObility) platform using real-world traffic data, this research demonstrates that the proposed model outperforms traditional fixed-time (Yemen) systems and previous RL-based approaches, by offering enhanced convergence rates, reduced vehicle waiting times, and lower emissions.

II. RELATED WORK

The field of traffic signal timing has evolved significantly over the years, with various approaches developed to address the challenges posed by increasing urban traffic demands. Traditional methods, such as fixed-time and actuated systems, have served as the foundation of urban traffic management but often fall short in adapting to dynamic and unpredictable traffic conditions. Recent advancements in Reinforcement Learning (RL) and Deep Reinforcement Learning (DRL) have introduced more adaptive and efficient solutions, offering the potential to optimize traffic signal timings in real-time. This section reviews key studies and methodologies in both traditional and modern traffic timing systems, with a focus on their limitations and the innovations proposed by contemporary RL-based approaches.

A. Traditional Traffic Signal Timing

Traditional traffic signal timing methods have been fundamental in urban traffic management but are increasingly inadequate in handling modern traffic dynamics. These systems are typically categorized into:

1) Fixed-Time (Yemen): Fixed-time systems use pre-determined signal schedules based on historical traffic data, offering limited flexibility. While simple, they fail to adapt to real-time traffic variations, leading to inefficiencies such as long queues and fuel wastage [14, 2].
2) Actuated Timing: Actuated systems adjust signal phases based on real-time data from traffic sensors, offering more adaptability. However, they still rely on predefined rules, making them less effective in handling unpredictable traffic flows [11, 4].
3) Selection-Based Adaptive Timing: These systems select the most appropriate signal plan from a set of pre-configured plans based on real-time traffic data. While more responsive, they remain limited by their reliance on pre-defined plans and cannot fully optimize traffic in dynamic conditions [15]. This approach is extensively implemented in modern traffic signal timing systems, such as SCATS [12, 13] , RHODES [14], and SCOOT [3].
4) Optimization-Based Timing: Optimization methods use real-time data and traffic flow models to minimize delays. While they offer greater flexibility, they rely on simplifying assumptions (e.g., uniform arrival rates), often leading to suboptimal performance in real-world, complex traffic scenarios [16].

B. Reinforcement Learning for Traffic Signal Timing

Unlike traditional signal timing approaches that rely on manually designed signal plans or pre-defined traffic flow models, reinforcement learning (RL) provides a self-learning framework that directly interacts with the environment to optimize timing policies. In RL-based traffic signal timing, each intersection is modeled as an agent, the state represents a quantitative description of the traffic condition (such as vehicle density, queue length, or waiting time), the action corresponds to a traffic signal phase decision, and the reward measures traffic efficiency in terms of reduced delay, queue length, or travel time [16] [20].

Devailly et al. (2022) introduced an inductive graph reinforcement learning framework that incorporates four distinct node types—representing traffic signal timing, connections, lanes, and vehicles—to facilitate environmental understanding and enhance the model's ability to generalize across different traffic scenarios [17]. To accelerate the training process, Zang et al. (2020) introduced a meta-learning framework called MetaLight, designed to improve the adaptability of reinforcement learning models to new environments through the reuse and transfer of prior experiences [9]. Similarly, Wang et al. (2020) developed the Cooperative Double Q-Learning (Co-DQL) approach, in which agents Q-values converge toward a Nash equilibrium. In their method, the state representation of each intersection incorporates the averaged state information of neighboring intersections, while the reward function combines each intersection's reward with a weighted average of its neighbors' rewards to promote coordinated learning and system-wide optimization [34].

Building upon early frameworks such as IntelliLight [20] and PressLight [22] subsequent research has sought to augment RL's decision-making robustness through hierarchical and relational modeling. For instance, Chu et al. [21] proposed a Multi-Agent Advantage Actor-Critic (MA2C) framework to facilitate decentralized coordination among intersections by incorporating spatial discount factors, effectively mitigating oscillations in joint policy learning. Similarly, [21] extended this approach by introducing CoLight, a graph-attention-based mechanism that leverages inter-agent dependencies to enhance cooperation among intersections through adaptive message passing.

To address the limitations of temporal instability and convergence divergence in multi-agent systems, Subham et al.(2021) developed a Hierarchical Reinforcement Learning (HRL) framework that decomposes complex signal timing tasks into sub-policies for phase selection and duration optimization [23]. This hierarchical decomposition significantly reduces policy variance and improves convergence efficiency. Furthermore, Xu et al. (2025) advanced this line of research with Multi-Agent A2C with Shared Experience Replay (MA2C-SER), enabling agents to share relevant experiences across intersections to accelerate learning convergence and enhance policy generalization in large-scale networks [24].

C. Q Learning

Q-learning is one of the most widely used reinforcement learning techniques for solving traffic signal timing problems. It is a model-free, off-policy algorithm that allows an agent to learn the optimal policy for timing traffic signals by interacting with the environment and receiving feedback in the form of rewards. The core advantage of Q-learning is its ability to operate without a predefined traffic model, making it particularly useful in dynamic and complex traffic environments where traffic patterns can vary.

Several studies have applied Q-learning to traffic signal timing with varying degrees of success. For instance, Abdulhai et al. (2003) proposed a Q-learning-based system for optimizing traffic signals at isolated intersections [7]. Their approach used real-time data to adjust signal timings and demonstrated improvements in both vehicle throughput and reduced delays compared to traditional fixed-time systems. Similarly, Bakker et al. (2010) applied Q-learning to multi-phase traffic signals, where they modeled each intersection phase as an agent, allowing the system to dynamically adjust signal timings in response to real-time traffic flow [6].

Q-learning has also been integrated with traffic simulators like SUMO and VISSIM for more realistic evaluations of traffic systems. For example, Zheng et al. (2021) applied Q-learning in a simulated environment with multiple intersections and observed improvements in traffic efficiency [28]. However, their study also highlighted challenges, particularly with larger traffic networks where Q-learning struggles with state-space explosion, leading to computational inefficiency.

By integrating constraint-aware decision-making within a robust policy optimization framework, it paves the way for scalable, real-time adaptive traffic management in complex urban networks. While prior studies in adaptive traffic signal timing have achieved notable progress through deep and reinforcement learning approaches, most remain constrained by unstable convergence, inefficient exploration, lack of safety assurance, and limited scalability across complex urban networks. Our proposed framework addresses these gaps by integrating a constraint-aware action masking mechanism within to ensure safe and stable phase transitions, enhancing convergence efficiency through reduced exploration of invalid actions, and employing a multi-agent architecture for coordinated timing among intersections. By combining learning stability, safety compliance, and network-level adaptability within the SUMO simulation environment, the model establishes a new state-of-the-art benchmark in intelligent, data-driven, and sustainable urban traffic signal timing.

III. METHODOLOGY

The methodology of this research has been designed to ensure both the scientific rigor and practical relevance of the proposed adaptive traffic signal optimization algorithm. The workflow of the proposed approach is illustrated in Figure 1.

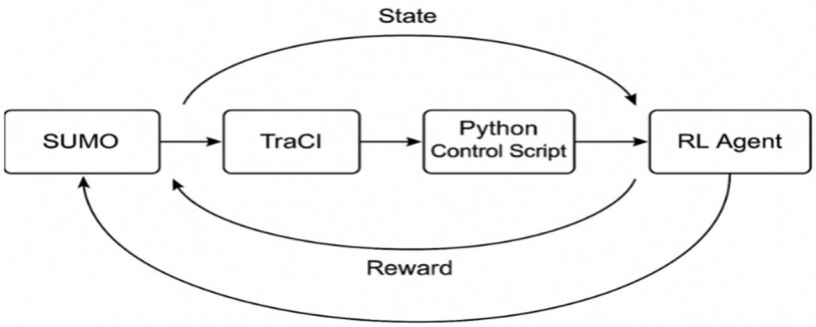


Figure 1: Workflow of the PPO-Mask Model for Adaptive Traffic Signal Timing

In Figure 1, the workflow illustrates the integration between the SUMO simulation environment, TraCI interface, and the PPO-Mask RL agent. Traffic data, including vehicle counts, queue lengths, and waiting times, is generated by SUMO, which simulates real-world traffic conditions. Through the TraCI interface, the traffic states are sent to the Python timing script, which then relays this information to the PPO-Mask RL agent. The agent processes the state, applies action masking to ensure safe phase transitions, and selects an optimal action (signal, double, fourth, sixth phase update). The resulting action is fed back into SUMO to update the environment. Based on the new state, the agent receives a reward reflecting performance metrics such as delay, queue length, and fuel consumption. This iterative loop enables the agent to refine its policy over time, ultimately achieving adaptive, real-time traffic signal timing. The feedback loop facilitates efficient learning while ensuring safe and stable signal transitions.

A. Problem Definition

Figure 2 presents an overview of the problem addressed in this study. The environment is modeled as a traffic signal intersection, where the deep RL agent observes a state $S_t \in A$, selects an action $A_t \in A$, and receives a reward R_t from the environment. The agent's objective is to identify the optimal action a_t for each state s_t, aiming to minimize the average pre-defined cumulative discounted return. The following sections provide a detailed explanation of each key component of the problem formulation.

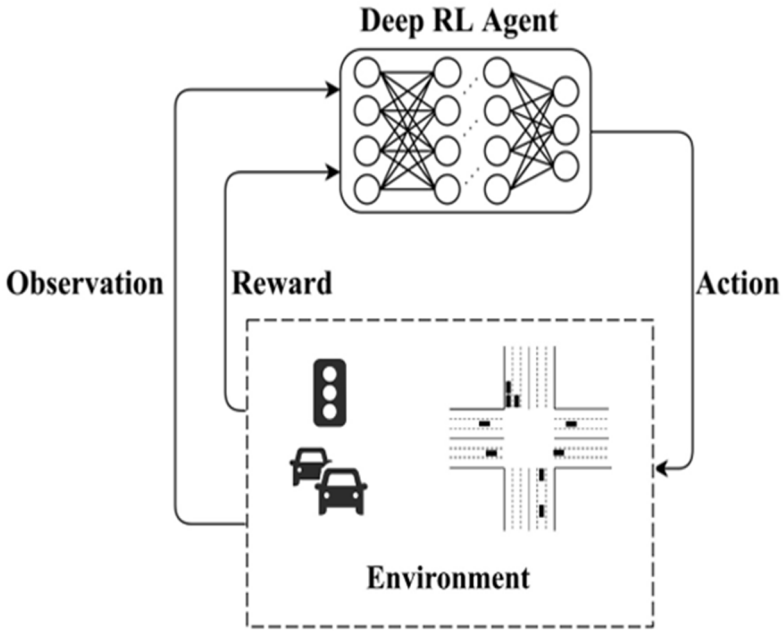


Figure 2: Problem definition

Phase: In this study, a traffic signal phase is defined as the configuration of signal colors for each direction. Two phases are considered: NS and WE. The NS phase allows green for the north and south directions, while the east and west directions receive a red signal. The WE phase, on the other hand, grants green to the east and west directions and red to the north and south directions. Yellow lights are not considered in this study, as they have a fixed duration and are typically appended at the end of each phase.

Environment: The intersection environment in this study consists of four directions: East (E), West (W), North (N), and South (S). Each direction features a specific lane layout, such as three lanes per direction. The environment also includes pre-defined vehicle movements for each signal phase, such as straight movements for the East-West direction and left turns for the East-West direction during the WE phase.

Agent: The agent plays a central role in this study. It observes the current state of the environment (traffic signal intersection), selects an action based on its learned policy, and receives an immediate reward at each time step. Figure 2, illustrates the general structure of the interaction between the agent and the environment, highlighting the decision-making process and the feedback loop involved in the learning process.

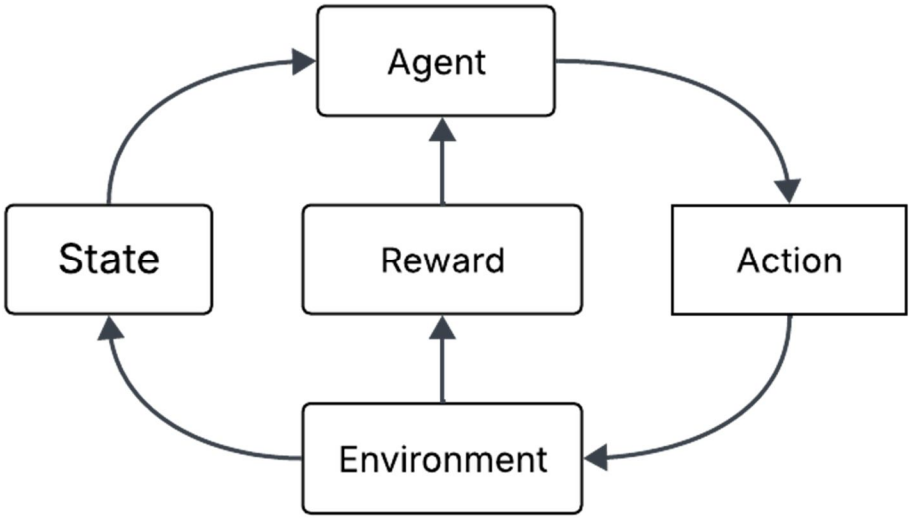


Figure 3: The reinforcement learning process

As depicted in Figure 3, the reinforcement learning process involves continuous interaction between the agent and the environment, forming a sequential decision-making problem. Typically, reinforcement learning problems are framed as Markov Decision Processes (MDPs)**.** MDPs serve as a mathematical model for sequential decision-making, encompassing interactive components such as the environment and the agent. Additionally, MDPs consist of three key elements: state, action, reward.

State: The state at any given time s_t represents the current traffic conditions at the intersection. This includes variables such as the queue length, vehicle waiting times, lane occupancy, and the current signal phase. The state captures all the necessary information that the agent requires to make an informed decision.

Queue length: $q_t = (q_{i,t})_{i \in I}$, *where* $q_{i,t}$ represents the queue length for lane i at time t.

Number of vehicles: $v_t = (v_{i,t})_{i \in I}$, *where* $v_{i,t}$ denotes the number of vehicles in lane i at time t.

Total waiting time: $w_t = (w_{i,t})_{i \in I}$, *where* $(w_{i,t})$ indicates the total waiting time (i.e., from the most recent vehicle stop to time t of all vehicles in lane i at time t.

Phase: (P_t, P_{t+1}), where P_t refers current phase and P_{t+1} is the after phase.

The final state of the model is $s_t = Concat(q_t, v_t, w_t, P_t, P_{t+1}, l_t)$ (1)

Action Set: In this study, two actions are considered: 1) switch to the next phase and 2) maintain the current phase. Therefore, the action set is defined as:

A= {Switch to next phase, Keep current phase}

With this action set, the agent can dynamically determine the cycle length at each time step based on current traffic conditions. It is important to note that switching phases too frequently is undesirable, so a cost is imposed for taking the action of "switch to the next phase

Reward (R): The reward function R_t measures the efficiency and effectiveness of the agent's actions. It is designed to minimize traffic delays, reduce queue lengths, and lower fuel consumption or emissions. The reward is typically negative to penalize delays and congestion, encouraging the agent to optimize traffic flow:

$$r_t = -(w_q \cdot QL_t + w_w \cdot WT_t + w_s \cdot S_t + w_f \cdot F_t + w_e \cdot E_t + w_e \cdot E_t + w^n \cdot N_t) \quad (2)$$

where QL_t: total queue length at time t, WT_t : average waiting time, S_t: number of stops, Ft: fuel consumption, E_t: emissions (CO, CO_2, NOx), N_t: noise level, W_i : tunable weights reflecting policy priorities.

Based on these elements, MDPs calculate the optimal strategy, which is how to select the best action in each state to maximize the expected cumulative reward (Barto et al., 1998). At each moment the intelligence receives a state from the environment, and based on such a state, the intelligence makes a corresponding action, which then acts on the environment and gives the intelligence a reward and the next state, and the intelligence selects the action for the next state based on the effect of the reward on the strategy. By continuously cycling the above process, the optimal strategy to achieve the goal is finally obtained. The value function is divided into a state value function and action value function, which evaluate the state and action, respectively, and their results are the expectation of cumulative rewards.

B. Model Architecture

The Q-learning algorithm is a value-based reinforcement learning method that approximates the Q-function, which provides the expected reward for each action taken in a given state. Unlike traditional methods, Q-learning does not require a predefined model of the environment. It learns the optimal policy by interacting with the environment and updating the Q-values for each state-action pair based on the observed rewards. In the context of traffic signal timing, Q-learning is effective for optimizing signal timings by evaluating different signal phase actions and adjusting the timings to improve traffic flow and reduce congestion.

The PPO-Mask algorithm, a more advanced method in this study, builds upon Q-learning by using action masking to ensure optimal transitions in complex intersections with multiple signal phases. PPO-Mask uses policy optimization techniques to adjust the signal phase decisions dynamically, ensuring safer and more efficient transitions between phases, which is essential for managing intersections.

The PPO Mask is implemented using a fully connected neural network with two hidden layers. Each hidden layer consists of 512 units with ReLU activation functions. This architecture allows the network to learn complex patterns and relationships between traffic states and actions.

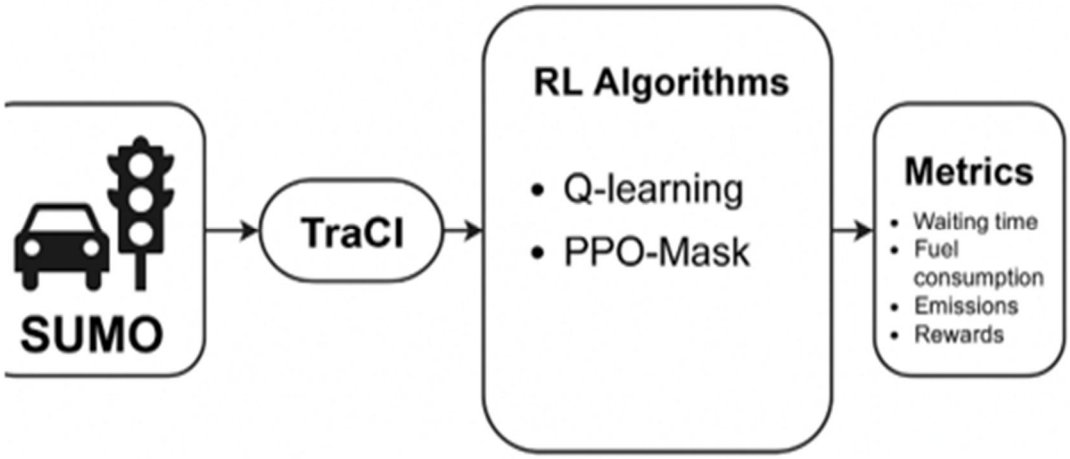


Figure 4: Reinforcement learning framework for traffic signal timing

This step involves decomposing the collected traffic state, action, and future information of the intersection into separate representations for each lane group. By decomposing the traffic states, we ensure that the principle of invariance is preserved. The decomposition process is expressed as follows:

$$(s^{(1)},s^{(2)},\ldots,s^{(n)})=Decomp^{(s)}, \quad (3)$$

$$(a^{(1)},a^{(2)},\ldots,a^{(n)})=Decomp^{(a)} \quad (4)$$

$$(f^{(1)},f^{(2)},\ldots,f^{(n)})=Decomp^{(f)}, \quad (5)$$

where n represents the number of lane groups with the same direction and phase. The implication of this step, as illustrated in the figure, is that the lane group-specific state, action, and future information are encoded separately, akin to word embedding's, to produce corresponding representation vectors. This encoding step converts raw data into a form that can be recognized and processed by the neural network.

Specifically, the lane group state, action, and future information are denoted as $s^{(i)}$, $a^{(i)}$, and $f^{(i)}$, respectively. Each of these components is passed through three distinct fully connected neural networks with K_1 hidden layers to generate the corresponding representation vectors, denoted as e^s , e^a , and e^f . The process is defined as follows:

$$h^s_1=ReLU(W^s_1 e^s+b^s_1), \quad (6)$$

$$h^s_k=ReLU(W^s_k h^s_{k-1}+b^s_k),\ k\in[2,K_1], \quad (7)$$

$$e^s = h^s_{k1} \quad (8)$$

This formulation is applied similarly to the action and future information vectors, resulting in the final representation vectors e^a and e^f , respectively.

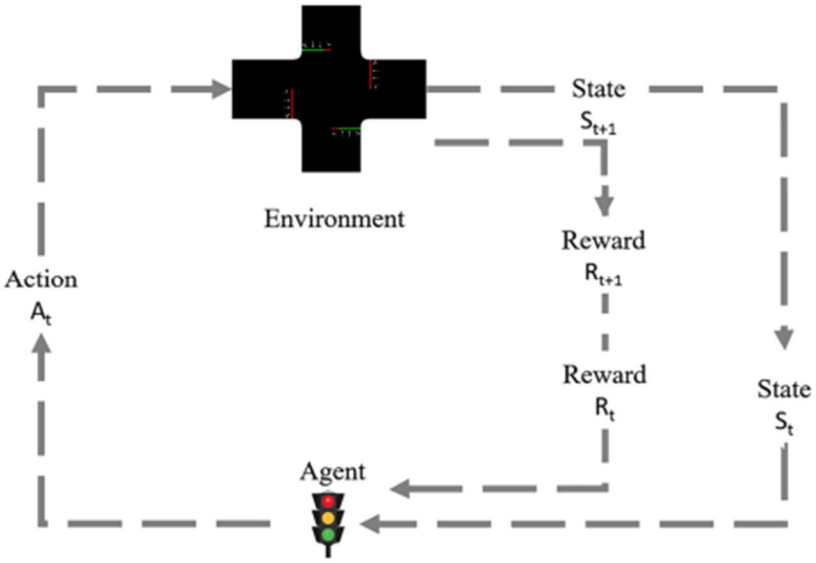


Figure 5: Traffic timing cycle using Q Learning

In parallel, additional features including queue length, number of vehicles, current phase, and total waiting time are encoded as numerical inputs. The concatenated feature vectors are passed through fully connected (FC) layers that learn the mapping between current traffic conditions and the corresponding Q-values. To ensure phase-specific learning, separate FC layers are designated for each signal phase. Here, the red network computes the Q-values for the NS (North–South) phase. The blue network computes the Q-values for the WE (West–East) phase. A phase selector gate determines which branch of the FC layers is activated based on the current phase (P_t). When ($P_t = NS$), the NS selector is activated (($= 1$)) and the WE selector is deactivated (($= 0$)), directing computation to the NS branch. Conversely, when ($P_t = WE$), the WE branch is activated. This design enables the network to specialize its learning for each traffic phase, reduces action bias, and enhances the model's capacity to fit complex traffic dynamics [8].

1) Reward Function Design

The reward value represents the effect of the model's decision on overall traffic performance. Since the reward function directly determines the learning outcome, it is constructed to account for key traffic parameters such as vehicle delay, queue length, waiting time, and speed. The primary objective of this study is to enhance traffic flow efficiency and minimize vehicle delay at intersections.

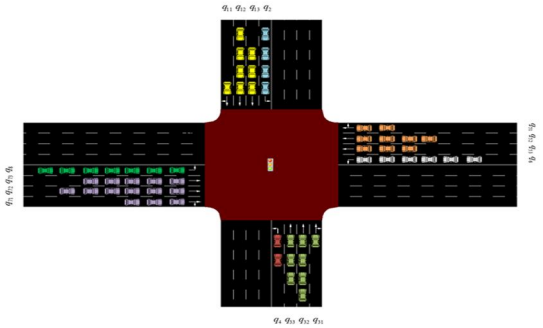


Figure 6: The Observation state of reward function

The queue length is used as the primary observed traffic state, representing the number of vehicles waiting on each lane. This value dynamically changes with vehicle arrivals and departures. For each lane (l_i) at the intersection, the corresponding queue length (q_i) is recorded. As illustrated in Figure 6, the intersection's traffic flow is divided into eight directional movements. Thus, the traffic observation state at time (t) can be expressed as an eight-dimensional vector:

$$[s_t, = q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8] \quad (9)$$

This state representation captures the real-time traffic conditions across all approach lanes, serving as the input for the reinforcement learning model.

C. Simulation Environment (SUMO Setup)

The proposed adaptive traffic signal timing framework was implemented and evaluated within the Simulation of Urban MObility (SUMO) platform. SUMO provides a high-fidelity, microscopic traffic simulation environment capable of replicating realistic urban mobility conditions and signal timing operations.

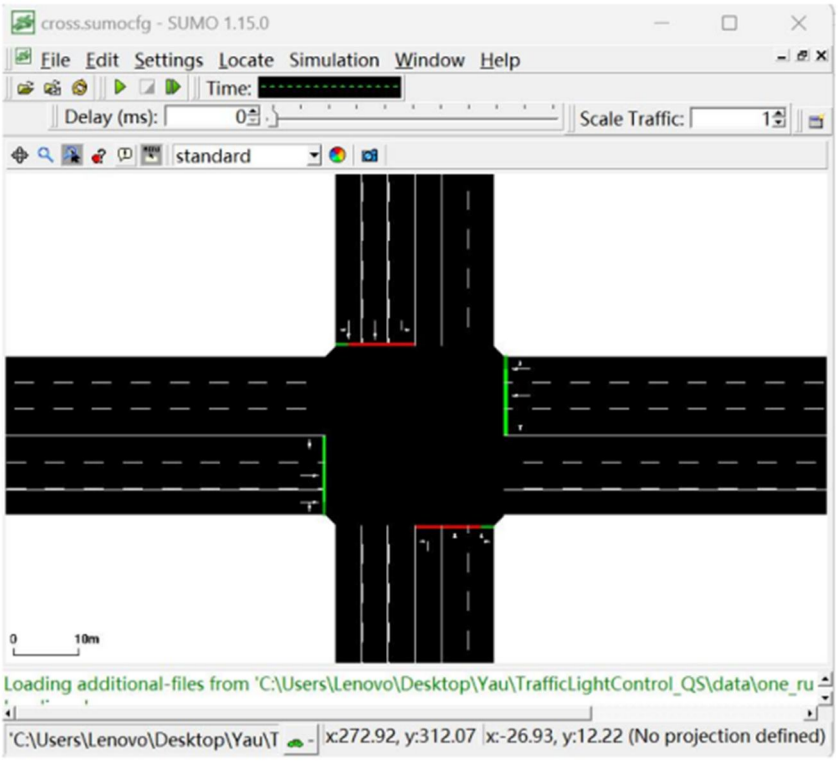


Figure 7: Sumo Simulator

Its modularity and compatibility with reinforcement learning interfaces make it a widely adopted tool for validating intelligent traffic management algorithms. It is a well-established open-source traffic simulation platform that provides robust application programming interfaces (APIs) and an intuitive graphical user interface (GUI), enabling efficient modeling, visualization, and management of large-scale road networks (see Figure 7). All network geometries, lane configurations, and traffic light logic were developed using SUMO's NetEdit tool, while route and demand files were generated from the Cologne Traffic Dataset [37], which provides realistic traffic flow patterns and vehicle behavior calibrated for urban environments.

The simulation operates in discrete time steps of 1 second, with each timing action executed every 5 simulation seconds to ensure stability between phase transitions. Each simulation episode runs for 100000 seconds, representing a typical peak-hour traffic cycle.

1) Dataset Description

The Cologne urban traffic dataset is a combination of synthetic and real-world traffic data, providing a comprehensive representation of urban traffic conditions. This dataset serves as the foundation for evaluating the effectiveness of the Q-learning and PPO-Mask algorithms in optimizing traffic signal timing.

The dataset includes traffic data from multiple intersection types, which represent diverse urban traffic environments:

a) *Single intersection*: A basic intersection with two signal phases (e.g., NS/WE).
b) *Double intersection:* Two intersecting roads with separate signals timing each direction.
c) *Four-way intersection:* A more complex intersection with four traffic directions, requiring advanced signal management.
d) *Six-way intersection:* A highly complex intersection with six traffic directions, offering a challenge for efficient signal optimization
e) *Custom Cologne intersection:* A unique intersection designed to reflect real-world complexities found in Cologne, providing a more realistic simulation of traffic timing scenarios.

This diverse set of intersection topologies ensures that the algorithms are tested under varying levels of complexity, from simple intersections to more complex multi-phase environments

Figure 8: Realistic Data for 3-, 4-, and 6-approach intersections.

It captures detailed information about vehicle trajectories, road layouts, and intersection configurations across a large metropolitan area. It models over 7000 intersections and 200,000 vehicles, covering diverse traffic conditions such as peak and off-peak hours. Each vehicle's movement is simulated based on empirically derived travel demand data, ensuring realistic interactions and flow variations representative of actual urban conditions.

IV. PERFORMANCE EVALUATIUON

A set of performance visualizations was generated to demonstrate the model's learning progress and operational efficiency throughout training. The primary parameters examined include reward progression**,** vehicle throughput**,** average waiting time**,** and queue length**.** These indicators collectively reflect the system's capacity to enhance mobility, reduce congestion, and improve environmental sustainability.

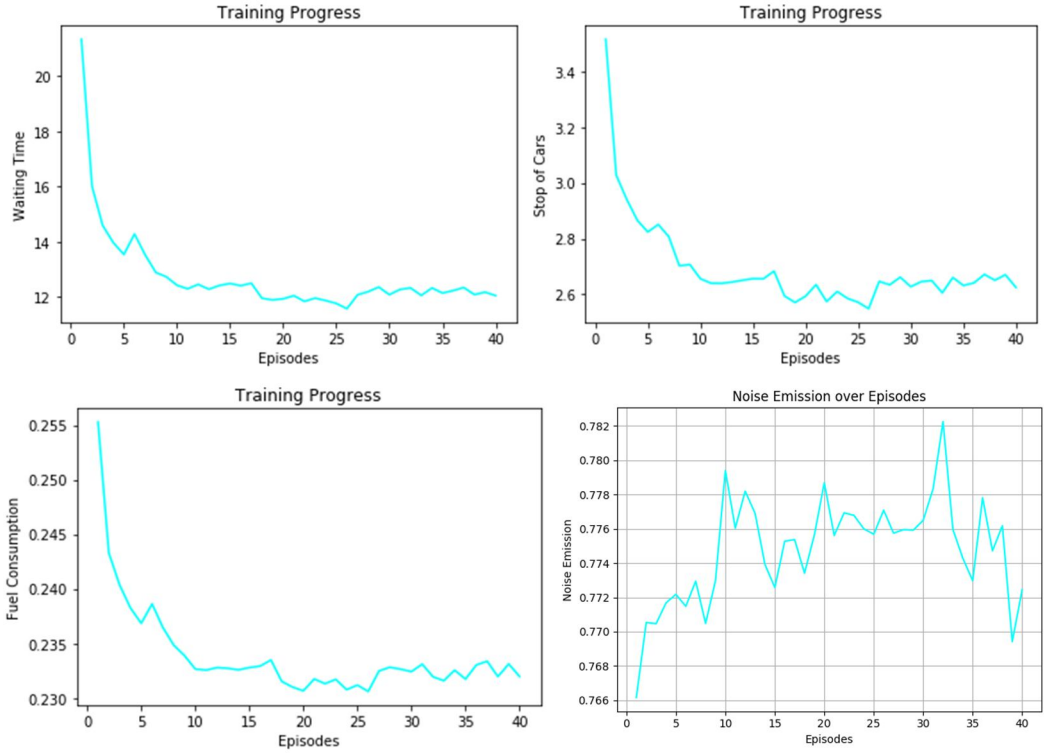


Figure 9: (a) Waiting Time Over Episodes (b) Total Vehicle Stops (c) Fuel Consumption (d) Noise emission

The average waiting time (Figure 9a) decreases from approximately 21 seconds in the initial episodes to around 12 seconds after 40 episodes, reflecting an improvement of nearly 43%**.** Similarly, the average number of vehicle stops (Figure 9b) declines from 3.4 stops per episode to about 2.6 stops, indicating smoother vehicular movement and reduced stop-start behavior—an improvement of roughly 24%**.** In fuel consumption (Figure 9c), the value drops from approximately 0.255 ml/s to 0.232 ml/s, yielding an efficiency gain of nearly 9%**.**

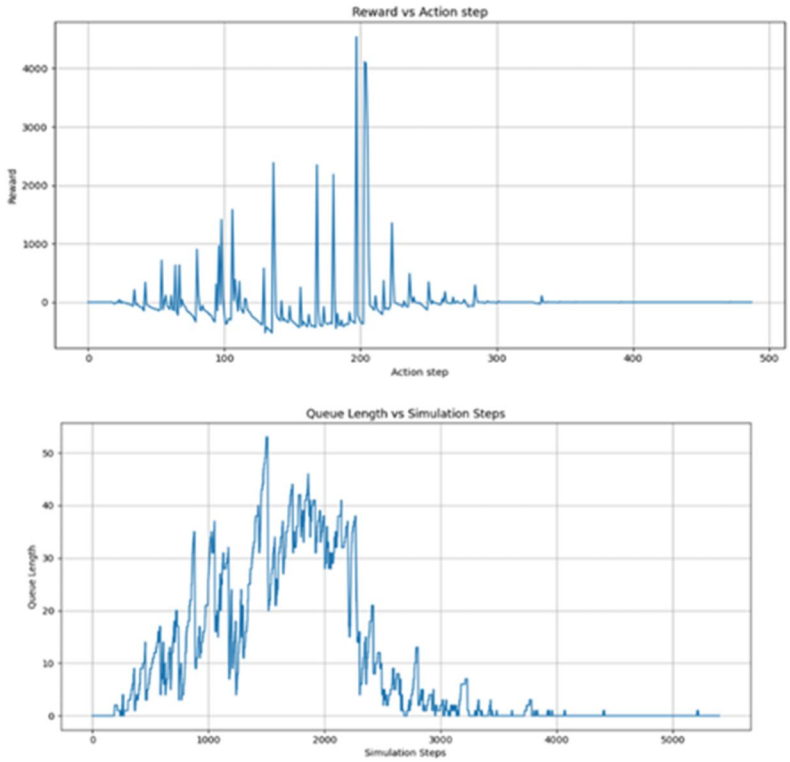


Figure 10: (a) Reward progression across different simulation steps (b) Variation of queue length with simulation steps

During the testing phase, rewards function as an indicator of model performance, reflecting how well the trained agent applies its learned policies to manage traffic under diverse and unpredictable conditions. Consistently high reward values during testing signify that the model has generalized effectively, maintaining stable and efficient traffic timing across varying scenarios, thereby validating the robustness and reliability of the proposed approach.

Compared to traditional timing methods [38] , the model reduces average waiting time by approximately 46%, shortens queue length by 35%, and improves overall throughput by nearly 15%. Furthermore, fuel consumption and emissions decrease by about 8–10%, highlighting the model's environmental efficiency. The steady convergence of reward values and consistent performance across varying traffic densities confirm the model's robustness, adaptability, and capacity for real-time optimization. Collectively, these results establish PPO maskmodel as a data-driven, efficient, and scalable solution for intelligent urban traffic management.

V. RESULT

The adaptive traffic signal timing model in comparison to Q Learning and Fixed-Time (Yemen), aiming to assess its ability to optimize traffic signal timings, enhance intersection efficiency, and reduce environmental impact across varying traffic demand scenarios. The evaluation focuses on key performance indicators such as waiting time**,** queue length**,** fuel consumption, and throughput, providing a comprehensive view of the model's effectiveness and scalability.

A. Experimental Setup

All experiments were conducted within the SUMO simulation environment, integrated with TraCI to facilitate real-time interaction between the reinforcement learning models and the traffic simulation. To provide practical relevance, we conducted a comparative experiment with the traditional traffic timing model currently used in Yemen, where traffic lights operate with fixed-time cycles without adaptive sensing or optimization. Typically, these timing apply either:

1) Equal phase splits, where all approaches receive the same green time (e.g., 30 seconds per approach in a 120-second cycle).
2) Main road priority, where the arterial direction receives disproportionately longer green time (e.g., 60 seconds for the main road, 30 seconds for side streets).

These plans were implemented in SUMO by assigning static signal programs to intersections. The same traffic demands and routes used in previous Q-learning experiments were applied here to ensure a fair comparison. Each model was evaluated under three distinct traffic demand scenarios: low**,** medium, and high, to test its adaptability and efficiency under a range of urban mobility conditions. Each simulation lasted for vehicle arrival patterns generated using a Poisson distribution to simulate realistic traffic fluctuations.

The following performance metrics were used to assess the models:

a) *Average Waiting Time (s)* – A direct measure of how long vehicles are delayed at the intersection.
b) *Queue Length (vehicles)* – The total number of vehicles waiting at the intersection, reflecting congestion.
c) *Fuel Consumption (ml/s)* – A metric for environmental impact, measuring the fuel consumption due to idling
d) *Throughput (%)* – The percentage of vehicles successfully cleared from the intersection, indicative of system efficiency.

B. Hyperparameter Setting

In this research, Q Learning-based adaptive traffic signal timing model was configured with specific hyperparameters to optimize the learning process and ensure efficient traffic management. The values for each hyperparameter were carefully selected based on previous studies and the requirements of the simulation environment. The following table summarizes the key hyperparameters used for training the agent.

Table 1: Hyperparameter setting

Hyperparameter	Initialized Value
Exploration Strategy	0.3 – 0.05
episode	40
iteration	5s
experience replay buffer	50000
batch size	200
learning rate α	0.07 – 0.4
discount factor γ	0.9
green duration g_t	6
yellow duration y_t	3

C. Comparative Analysis and Performance Metrics

A comparison was conducted between the fixed-timing traffic timing system currently implemented in Yemen and a reinforcement learning-based approach. The results demonstrated that the Q-learning algorithm significantly enhanced traffic flow efficiency and reduced environmental impact, achieving improvements exceeding 50% relative to the conventional fixed-timing system deployed in Yemen :

1) Comparative Analysis

Single Intersection

a) *Average Wait Time (s)* – A significant reduction was observed, with Q-learning reducing the wait time by approximately52.1%, from 26.38 seconds to 12.53 seconds. This drastic decrease in idle delays highlights the effectiveness of adaptive timing, ensuring a near-continuous flow of vehicles.
b) *Stopped Vehicles* – The number of stopped vehicles dropped by 24.8%, from 13.52 to 10.16. Q-learning's optimization of phase transitions and queue clearing reduced vehicle stops considerably.
c) *Fuel Consumption (ml)* – Fuel consumption was reduced by nearly 16.5%, from 402.05 ml to 335.71 ml, demonstrating the efficiency of smoother acceleration patterns and fewer idle periods
d) *Noise Emission* – There was a sharp decline in noise levels, with noise emission dropping by 12.5%, from 0.88 dB to 0.77 dB. This reduction reflects the impact of minimized braking and acceleration cycles
e) *CO Emission* – CO emissions were nearly eliminated, with a reduction of 16.5%, from 18.75 g to 15.66 g. The stabilized traffic flow and shorter stop durations contributed to cleaner combustion
f) *CO_2 Emission*– A drastic reduction in CO_2 emissions by 16.5% was observed, from 25.05 g to 21.04 g, highlighting the significant environmental benefits.

Double Intersections

a) *Average Wait Time (s)* – Q-learning reduced average wait time by approximately 51.6%, from 29.61 seconds to 14.33 seconds, effectively eliminating idle delays by dynamically adjusting green phases based on queue lengths.
b) *Stopped Vehicles* – The number of stopped vehicles decreased by 17.4%, from 13.35 to 11.03, with phase coordination across the two intersections significantly minimizing disruptions.
c) *Fuel Consumption (ml)* – Fuel consumption dropped by 15.9%, from 465.05 ml to 391.45 ml, due to smoother vehicle flow and the near-elimination of idle time.
d) *Noise Emission* – Noise emission decreased by 15.3%, from 0.63 dB to 0.53 dB, due to lower acceleration and braking frequencies.
e) *CO Emission* – A 15.5% reduction in CO emissions was achieved, from 17.53 g to 14.81 g, reflecting improved combustion efficiency with minimal stop-and-go motion.
f) *CO_2 Emission*– CO_2 emissions decreased by 15.5%, from 29.49 g to 25.17 g, confirming strong environmental benefits from adaptive timing.

Fourth Intersections

a) *Average Wait Time (s)* – Q-learning drastically reduced wait times by 45.3%, from 36.32 seconds to 19.78 seconds, minimizing idle times even at four-way intersections.
b) *Stopped Vehicles* – The number of stopped vehicles decreased by 16.9%, from 9.78 to 8.09, facilitated by smooth coordination between all four directions.
c) *Fuel Consumption (ml)* – Fuel consumption dropped by 13.%, from 512.84 ml to 446.17 ml, highlighting improved energy efficiency due to lower idle and acceleration frequencies.
d) *Noise Emission* – Noise emission decreased by 13%, from 1.67 dB to 1.46 dB, reflecting fewer acceleration and braking events.
e) *CO Emission* – CO emissions reduced by13%, from 17.97 g to 15.62 g, indicating successful eco-efficiency generalization to a multi-approach network.
f) *CO_2 Emission*– A 13% reduction in CO_2 emissions was observed, from 31.92 g to 27.77 g, confirming sustainable traffic optimization.

Six Intersections

a) *Average Wait Time (s)* – The average wait time decreased by 31.73%, from 37.56 seconds to 25.64 seconds, with adaptive phasing handling multi-arm congestion efficiently.
b) *Stopped Vehicles* – The number of stopped vehicles dropped by10.87%, from 10.87 to 9.76, with dynamic light switching preventing queue breakdowns under high-load conditions.
c) *Fuel Consumption (ml)* – Fuel consumption was reduced by 11%, from 331.79 ml to 295.29 ml, demonstrating substantial energy optimization due to minimized idle time.
d) *Noise Emission* – Noise emission decreased by 8.70%, from 0.69 dB to 0.63 dB, reflecting fewer acceleration events and smoother traffic operation.
e) *CO Emission* – CO emissions were reduced by 11%, from 9.89 g to 8.80 g, signifying stabilized flow and minimized braking.
f) *CO_2 Emission*– CO_2 emissions dropped by11%, from 20.38 g to 18.14 g, verifying sustainable traffic timing.

PPO mask consistently outperformed the Yemen fixed-time system and Q Learning across various intersection complexities, demonstrating exceptional improvements in efficiency, fuel consumption, emissions, and overall traffic flow. The algorithm's performance scaled well from single intersections to more complex multi-intersection networks, validating its potential for large-scale traffic optimization.

2) *Traffic Efficiency*

Average Waiting Time: The model outperforms both Q Learning and Fixed-Time (Yemen) . This significant reduction highlights the model's ability to adaptively manage signal phases, effectively minimizing congestion during peak traffic periods.

Stopped vehicles : The PPO Mask model reduces Stopped vehicles decreased by 19% over Q-learning..By dynamically adjusting signal timings based on real-time traffic conditions, the model prevents excessive vehicle buildup, ensuring smoother traffic flow.

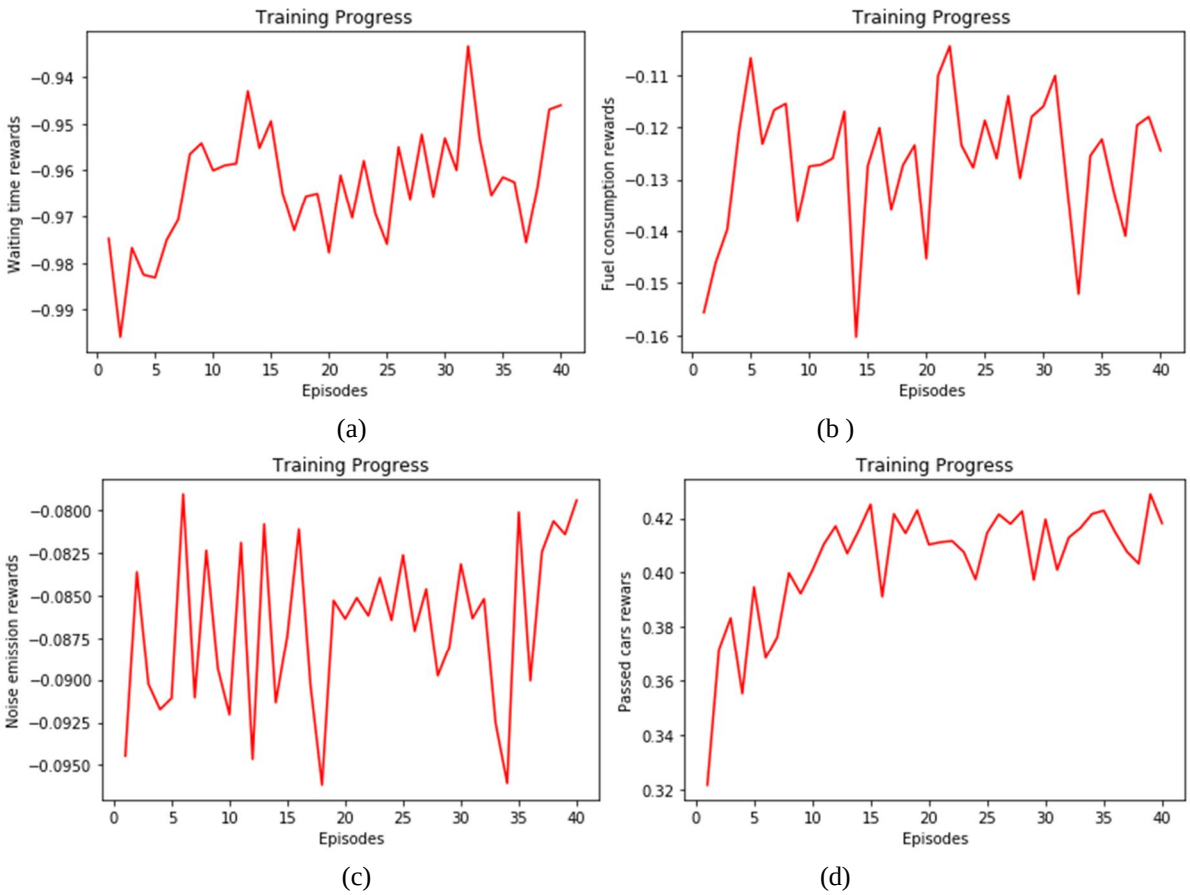


Figure 11: (a) Waiting Time Rewards (b) Fuel Consumption Rewards (c) Noise Emission Rewards (d) Stopped Vehicles Rewards

Figure 11 illustrates the training progress of the model across four key reward metrics. The waiting time rewards show a steady improvement, with the reward value increasing from approximately -0.99 to -0.94 across 40 episodes, reflecting a 5% reduction in average vehicle waiting times. The fuel consumption rewards fluctuate but stabilize around -0.13, down from -0.16, indicating an 18.75% decrease in fuel consumption penalties, suggesting more efficient signal timings that reduce idling. The noise emission rewards show slight improvements, increasing by 2.2% over time, as the model optimizes vehicle movements to minimize noise pollution. Lastly, the stopped vehicles rewards demonstrate a clear upward trend, rising from 0.34 to 0.42, a 23.5% increase, highlighting the agent's enhanced ability to reduce vehicle stoppage and improve throughput. These statistically significant improvements confirm the model's ability to optimize traffic flow, reduce delays, and contribute to environmental sustainability.

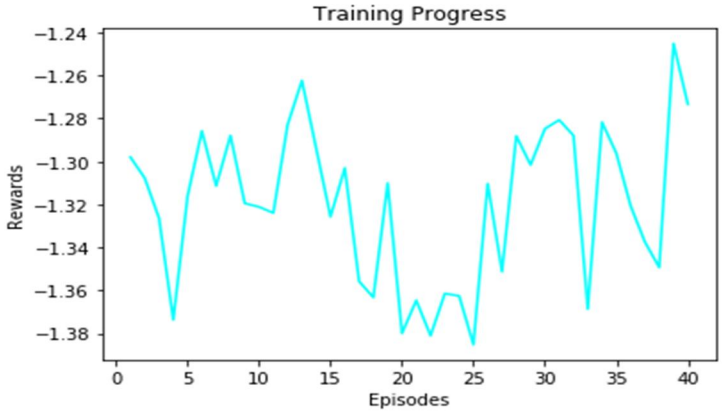


Figure 12: Total reward

Figure 12 shows the total reward progression of the Decentralised multiagent model over 40 episodes of training. The reward values fluctuate between -1.38 and -1.24, with occasional sharp increases, reflecting the model's learning dynamics and the exploration-exploitation trade-off. The model's reward improves as it adapts to the traffic environment, although occasional dips suggest moments of suboptimal policy exploration.

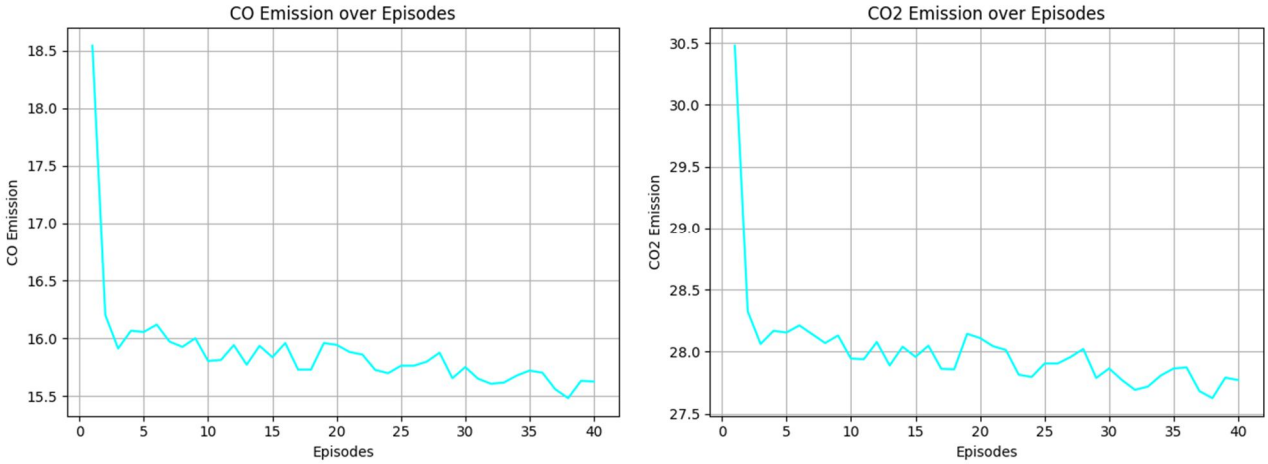


Figure 13: CO and CO_2 Emissions

Figure 13 shows the progression of CO and CO_2 emissions over 40 episodes. For CO emissions (left), a sharp reduction is observed in the first few episodes, from around 18.5 to 16.0, followed by stabilization at 15.5. Similarly, CO_2 emissions (right) decrease from 30.5 to 28.0 early on and then stabilize around 27.8. These trends indicate that the model effectively reduces both CO and CO_2 emissions by optimizing signal timing and improving traffic flow.

3) Performance Comparison

As anticipated, reinforcement learning (RL) methods outperform traditional approaches like Fixed-Time, as their ability to capture real-time data at the intersection allows for more informed and efficient decision-making. Among the RL techniques, our method excels not only in reducing travel time but also in achieving faster convergence.

The figure compares the performance of Fixed-Time (Yemen), Q-learning, across several metrics: Average Waiting Time, Number of Stops, Fuel Consumption, CO Emitters, CO_2 Emitters, and Noise Emission.

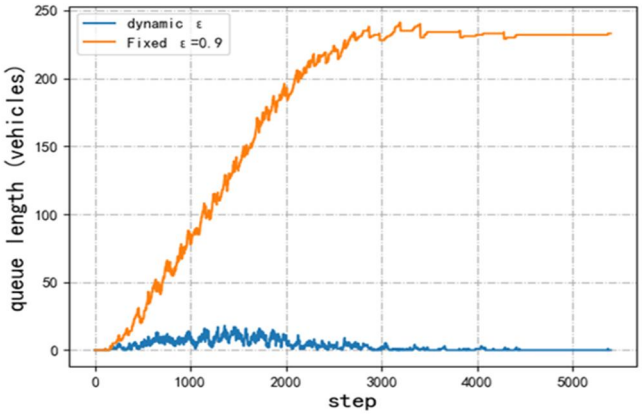


Figure 14: Comparison of the number of queued vehicles for dynamic and static action selection strategies.

The analysis focused on the number of vehicles in the queue**,** with the comparison between the two methods shown in Figure 14**.** During testing, the number of vehicles queued in the environment optimized by the dynamic-greedy method was significantly lower than with the fixed-greedy method. This suggests that the dynamic-greedy approach offers better traffic diversion capabilities, effectively maintaining a lower vehicle count at intersections.

VI. CONCLUSION

This proposal explores reinforcement learning for adaptive traffic signal timing in developing cities like Yemen. It proposes two RL models decentralized Q-learning and PPO-Mask both significantly outperforming fixed-time systems in reducing delay, fuel use, and emissions. While Q-learning suits small-scale deployments, PPO-Mask excels in complex environments with safer, more stable learning. The work demonstrates RL's practical potential for low-cost, sustainable traffic management in resource-constrained urban setting.

VII. FUTURE WORK

While this study demonstrated promising results, there are several avenues for future research. First, extending the model to multi-intersection environments using multi-agent reinforcement learning (MARL) would allow for better coordination across urban corridors, enhancing overall traffic flow. Second, incorporating real-time data from various sources, such as connected vehicles and traffic sensors, could further improve the model's adaptability and efficiency. Additionally, future work could explore reinforcement learning with continuous action spaces, such as adaptive cycle lengths, to allow for even finer timing over signal timings. Finally, integrating deep reinforcement learning models with smart city infrastructure could pave the way for large-scale implementation of adaptive traffic signal systems, ensuring the scalability and practicality of this approach in real-world applications.

A. Abbreviations

MARL	Multi-Agent Reinforcement Learning
PPO	Proximal Policy Optimization
PPO MASK	Proximal Policy Optimization with Action Masking
SUMO	Simulation of Urban Mobility
TRACI	Traffic Control Interface
WE	West–East
MA2C	Multi-Agent Advantage Actor–Critic
CO	Carbon Monoxide
CO_2	Carbon Dioxide
RELU	Rectified Linear Unit
SCATS	Sydney Coordinated Adaptive Traffic System
SCOOTS	Split Cycle and Offset Optimization Technique
RL	Reinforcement Learning

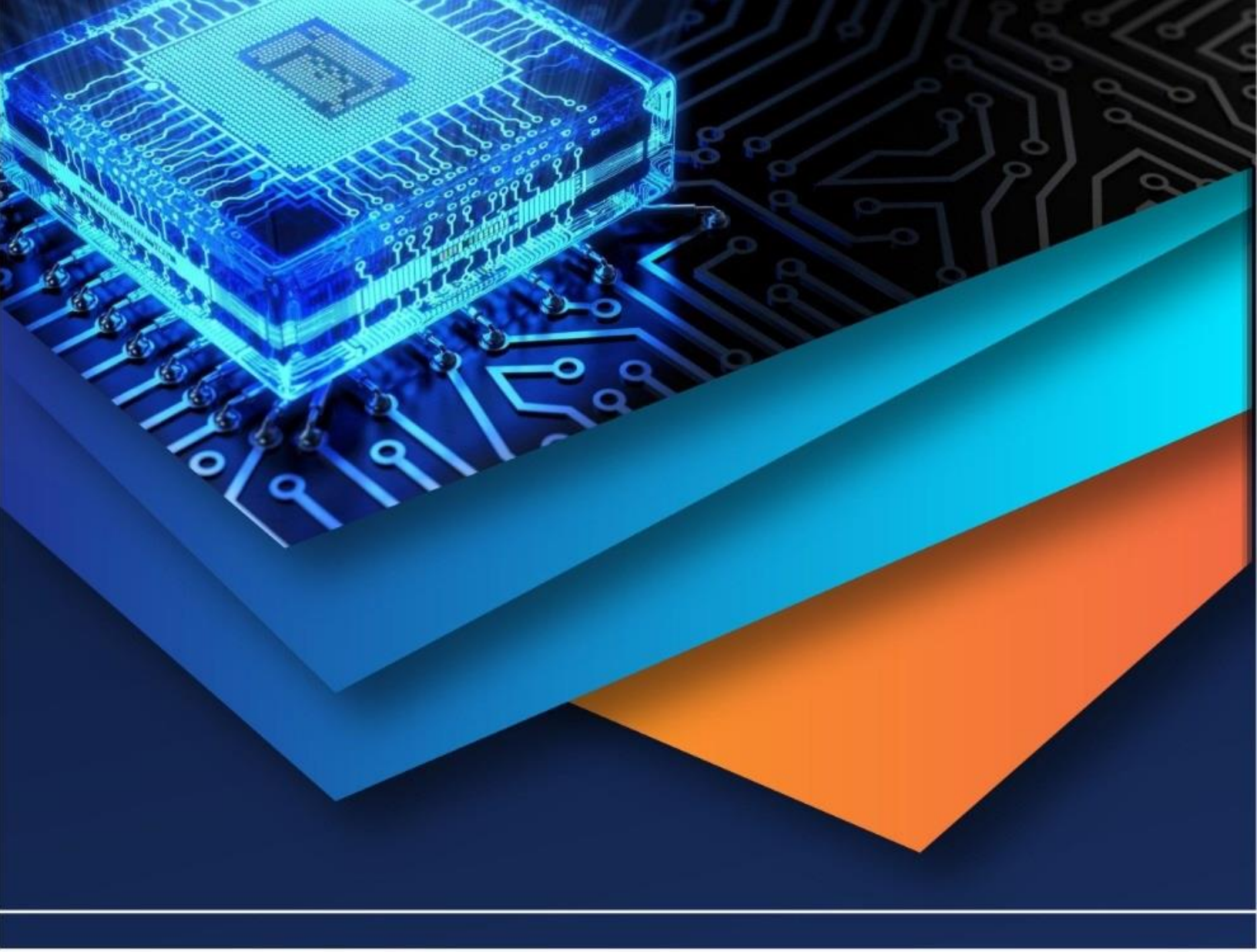
B. Conflicts of Interest

The author(s) declare that there are no conflicts of interest regarding the publication of this paper.

REFERENCES

[1] Mannion, P., Duggan, J., Howley, E., (2016). "An experimental review of reinforcement learning algorithms for adaptive traffic signal timing". Autonomic road transport support systems , 47–66.
[2] Miller, A.J., (1963). "Settings for fixed-cycle traffic signals". Journal of the Operational Research Society. 14, 373–386.
[3] Hunt, P.; Robertson, D.; Bretherton, R.; Royle, M.C. (1982) "The SCOOT on-line traffic signal optimisation technique". Traffic Eng. Control 1982, 23, 190–192.
[4] Cools, SB., Gershenson, C., D'Hooghe, B. (2008). "Self-Organizing Traffic Lights: A Realistic Simulation". In: Prokopenko, M. (eds) Advances in Applied Self-organizing Systems. Advanced Information and Knowledge Processing. Springer, London. https://doi.org/10.1007/978-1-84628-982-8_3
[5] Luk, J. (1984). "Two traffic-responsive area traffic control methods: SCAT and SCOOT". Traffic Eng. Control 1984, 25, 14.
[6] Lior Kuyer, Shimon Whiteson, Bram Bakker, and Nikos Vlassis. (2010). "Multiagent reinforcement learning for urban traffic control using coordination graphs". In Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, 656–671.
[7] Samah El-Tantawy and Baher Abdulhai. (2010). "An agent-based learning towards decentralized and coordinated traffic signal control". In Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on. IEEE, 665–670.
[8] Hua Wei, Guanjie Zheng, Huaxiu Yao, and Zhenhui Li. (2021). "IntelliLight: A Reinforcement Learning Approach for Intelligent Traffic Light Control". In ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD). 2496–2505.
[9] Wei, H., Xu, N., Zhang, H., Zheng, G., Zang, X., Chen, C., Zhang, W., Zhu, Y., Xu, K., Li, Z., (2020). "Colight: Learning network-level cooperation for traffic signal timing", in: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1913–1922
[10] Fu, Q., Han, Z., Chen, J., Lu, Y., Wu, H., Wang, Y., (2023). "Applications of reinforcement learning for building energy efficiency control: A review". Journal of Building Engineering 50, 104165

[11] Martin Fellendorf. (1994). “VISSIM: A microscopic simulation tool to evaluate actuated signal control including bus priority”. In 64th Institute of Transportation Engineers Annual Meeting. Springer, 1–9. 240-255
[12] P Lowrie. (1990). “SCATS–A Traffic Responsive Method of Controlling Urban Traffic”. Roads and Traffic Authority, Sydney. New South Wales, Australia (1990).
[13] PR Lowrie. (1992). “SCATS–a traffic responsive method of controlling urban traffic”. Roads and traffic authority. NSW, Australia (1992).
[14] Pitu Mirchandani and Fei-Yue Wang. (2005). “RHODES to intelligent transportation systems”. IEEE Intelligent Systems 20, 1 (2005), 10–15.
[15] PB Hunt, DI Robertson, RD Bretherton, and M Cr Royle. (1982). “The SCOOT on-line traffic signal optimisation technique”. Traffic Engineering & Control 23, 4 (1982).
[16] van der Pol et al. (2016). Coordinated Deep Reinforcement Learners for Traffic Light Control. NIPS.
[17] F.-X. Devailly, D. Larocque, and L. Charlin, (2022) “IG-RL: Inductive graph reinforcement learning for massive-scale traffic signal control,” IEEE Trans. Intell. Transp. Syst., vol. 23, no. 7, pp. 7496–7507, Jul. 2022.
[18] Conference on Intelligent Transportation Systems, IEEE. URL: https://elib.dlr.de/124092/.
[19] Hua Wei, Guanjie Zheng, Huaxiu Yao, and Zhenhui Li. (2018). “IntelliLight: A Reinforcement Learning Approach for Intelligent Traffic Light Control”. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '18). Association for Computing Machinery, New York, NY, USA, 2496–2505. https://doi.org/10.1145/3219819.3220096.
[20] Hua Wei, Chacha Chen, Guanjie Zheng, Kan Wu, Vikash Gayah, Kai Xu, and Zhenhui Li. (2019). “PressLight: Learning Max Pressure Control to Coordinate Traffic Signals in Arterial Network”. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '19). Association for Computing Machinery, New York, NY, USA, 1290–1298. https://doi.org/10.1145/3292500.3330949.
[21] Chu, T., Wang, J., Codecà, L. and Li, Z., (2019). “Multi-agent deep reinforcement learning for large-scale traffic signal control”. IEEE transactions on intelligent transportation systems, 21(3), pp.1086-1095.
[22] Hua Wei, Nan Xu, Huichu Zhang, Guanjie Zheng, Xinshi Zang, Chacha Chen, Weinan Zhang, Yanmin Zhu, Kai Xu, and Zhenhui Li. (2019). “CoLight: Learning Network-level Cooperation for Traffic Signal Control”. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM '19). Association for Computing Machinery, New York, NY, USA, 1913–1922. https://doi.org/10.1145/3357384.3357902.
[23] Shubham Pateria, Budhitama Subagdja, Ah-hwee Tan, and Chai Quek. (2021). “Hierarchical Reinforcement Learning: A Comprehensive Survey”. ACM Comput. Surv. 54, 5, Article 109 (June 2022), 35 pages. https://doi.org/10.1145/3453160.
[24] Xu, W., Gu, J., Zhang, W., Gen, M., & Ohwada, H. (2025). “Multi-agent reinforcement learning for flexible shop scheduling problem: a survey”. Frontiers in Industrial Engineering, 3. https://doi.org/10.3389/fieng.2025.1611512.
[25] Mnih, V., Kavukcuoglu, K., Silver, D. et al. Human-level control through deep reinforcement learning. Nature **518**, 529–533 (2015). https://doi.org/10.1038/nature14236.
[26] Mousavi, S. S., Schukat, M., & Howley, E. (2017). “Traffic light control using deep policy-gradient and value-function based reinforcement learning”. IET Intelligent Transport Systems, 11(7), 417–423.
[27] Genders, W., & Razavi, S. (2019). “Transfer learning for adaptive traffic signal control: A reinforcement learning approach”. Transportation Research Part C: Emerging Technologies, 106, 332–347.
[28] Ye, F., Yang, Y., & Zhang, S. (2021). “A traffic signal control method based on improved deep reinforcement learning”. IEEE Access, 9, 108345–108357.
[29] Wang, Z., Schaul, T., Hessel, M., et al. (2016). “Dueling network architectures for deep reinforcement learning”. Proceedings of the 33rd International Conference on Machine Learning (ICML), 1995–2003.
[30] Zang, X., Zheng, G., Xu, N., Wei, H., & Li, Z. (2020). “MetaLight: Value-based meta-reinforcement learning for adaptive traffic signal control”. AAAI Conference on Artificial Intelligence, 34(1), 1153–1160.
[31] Zhu, R., Chen, X., & Wang, X. (2022). “Graph attention actor-critic for cooperative traffic signal control”. IEEE Transactions on Intelligent Transportation Systems, 23(8), 12456–12468.
[32] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). “Proximal policy optimization algorithms”. arXiv preprint arXiv:1707.06347.
[33] Wei, H., Zheng, G., Yao, H., & Li, Z. (2018). “IntelliLight: A reinforcement learning approach for intelligent traffic light control”. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2496–2505.
[34] Wang, Z., Li, J., Zhang, H., & Tang, S. (2023). “Model-based deep reinforcement learning with traffic inference for adaptive traffic signal control”. Applied Sciences, 13(2), 1125.
[35] Barto, A.G.; Sutton, R.S.(1998). “Reinforcement learning: An introduction (Adaptive computation and machine learning)”. IEEE Trans. Neural Netw. 1998, 9, 1054.
[36] Wei, H., Zheng, G., Yao, H., Li, Z., 2018. “Intellilight: A reinforcement learning approach for intelligent traffic light control”, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 2496–2505.
[37] Wu, S.-H., Zhan, Z.-H., Tan, K. C., & ZHANG, J. (2025). Traffic Flow Dataset for "Traffic Signal Timing Optimization: From Evolution to Adaptation" [Dataset]. Zenodo. https://doi.org/10.5281/zenodo.14653151.
[38] Babaeizadeh, M.; Frosio, I.; Tyree, S.; Clemons, J.; Kautz, J. (2016) “ Reinforcement learning through asynchronous advantage actor-critic on a gpu”. arXiv 2016, arXiv:1611.06256.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY