A Novel Edge-Enabled Adaptive Traffic Light System Using Nonlinear Optimization

Zainab Kadhim Abed
Department of Computer Engineering, Al-Nahrain University, 10072 Baghdad, Iraq
st.compe.zainab.k.abed@ced.nahrainuniv.edu.iq

Keywords: Edge Computing, Adaptive Traffic Signal Control (ATSC), NMATCS, Deep Q-Network (DQN), Internet of Things (IoT).

Abstract: Urban traffic congestion remains a critical challenge due to the exponential growth of automobiles, rapid urbanization, and escalating demand for transportation, all of which strain global road infrastructure. Traditional traffic light systems and recent baseline technologies have proven inefficient under dynamic conditions. To address this, we propose an Adaptive Traffic Signal Control (ATSC) system designed to alleviate urban congestion by adjusting signal timing in real-time based on fluctuating demand. Deep Q-learning Network (DQN) recent approaches begin with no previous knowledge and display substantial unpredictability in action selection, which can lead to poor control effects initially. This study proposes an efficient traffic prediction model based on the integration of edge computing near to IoT sensors to ensure accurate real-time information. The proposed Nonlinear Mathematical Adaptive Traffic Light System (NMATCS) algorithm employs an exponential function to dynamically modify traffic signal duration based on road vehicle density, ensuring no time is wasted at any signal phase. The focus of this study is to reduce the Average Waiting Time (AWT) across multiple intersections. The (NMATCS) algorithm was tested in a simulation environment using authentic real traffic data from four intersections in Hangzhou, China, during peak hours. Performance was benchmarked against the state-of-the-art Priority-based Double Deep Q-Learning (Pri-DDQN) method. Our proposed model achieved an (AWT) of 25.2 seconds, outperforming Pri-DDQN (26.0 seconds) at isolated intersections and DQN-Baseline (28.3 seconds), with improvements of 3.08% and 10.95%, respectively. The results demonstrate that edge computing-enhanced IoT architectures ensure a smooth, nonlinear response and effectively address latency, scalability, and real-time decision-making limitations inherent in cloud-dependent systems. The findings highlight the potential of lightweight, edge-enabled adaptive traffic control in improving urban mobility without relying on computationally intensive learning techniques.

1 INTRODUCTION

1.1 Edge Computing in the (IoT) Vision

The Internet of Things (IoT) [1] aims to integrate the digital world with the real world; however, as the volume of data and devices increases, restrictions arise due to a lack of accuracy, delays in data transfer, computing delays, and high costs. Thus, merging it with peripheral computing will improve data accuracy and transfer speed. Edge computing is one of the essential technologies, particularly important for managing an adaptive traffic system that regulates and controls the flow of transportation. The Internet of Things (IoT) has emerged as a transformative technology for efficient traffic management, facilitating the accumulation, analysis, and automated decision-making of real-time data to optimize urban mobility [2]. Researchers implemented an IoT-enabled adaptive traffic signal system that resulted in a 21% reduction in average travel time and a 20% reduction in emissions. To detect congestion, monitor traffic flow, and dynamically alter signal timings, IoT-based traffic systems incorporate a network of connected devices, cameras, and sensors, including inductive loops, LiDAR, and GPS-enabled motor vehicles [3]. Recent developments suggest combining edge computing with IoT to solve these issues by allowing localised data processing at traffic crossings, lowering latency and increasing responsiveness. Studies have shown that edge-enabled (IoT) traffic systems can reduce AWT by up to 40% when compared to conventional systems [4]. Edge computing-enabled smart cities have seen a 40% decrease in Edge computing has made possible new applications that were previously unattainable with conventional cloud architectures, as seen by recent implementations reported in emergency response times and a 55% increase in traffic management effectiveness. Edge computing-enabled industrial IoT deployments have shown a 50% increase in predictive maintenance accuracy and a 65% reduction in quality control problem [5]. (IoT) has revolutionized the way data is processed and analyzed, with edge computing and cloud computing emerging as two distinct paradigms. While both technologies are integral to IoT ecosystems, what is Edge Computing? Edge computing pertains to the technologies that facilitate computational processes at the periphery of the network, utilizing downstream data in support of cloud services and upstream data on behalf of Internet of Things (IoT) services. In this context, the term "edge" is defined as any computing and networking resources situated along the route connecting data sources to cloud data centers [6]. Figure 1 displays edge computing Benefits.

1.2 Edge as a Promising Alternative to Cloud in Traffic Management

Rapid urbanization and increased vehicular traffic necessitate the use of intelligent traffic management systems (ITMS) to minimize congestion, while computing has historically been the backbone of ITMS. A promising solution to the limitations of cloud computing in ITMS has emerged in the form of edge computing. This technology is capable of reducing latency and bandwidth utilization by processing data closer to the source, rendering it suitable for real-time traffic management applications [7]. The fundamental architectural difference between cloud and edge paradigms is illustrated in Figure 2. Real-time data processing from sensors, cameras, and IoT devices fuels traffic management systems' optimised signal control, route planning, and issue detection. While edge computing handles data closer to the source, hence lowering latency and bandwidth usage, cloud computing provides centralized processing with great computational capability [8].

1.3 An Over View of Control Traffic Light Systems

With the development of smart city infrastructures, the Internet of Things has become widely employed in intelligent transportation systems, such as data collecting, information processing, and traffic light management Smart cities use IoT sensors to collect data, manage resources, and deliver public services. An excessive number of vehicles moving from one location to another and the lack of proper flow management disrupts the smooth traffic flow. Resulting in terrible traffic congestion and inappropriate traffic congestion [9].

Multiple methodologies exist to detect traffic congestion, including image analysis, laser tracking, and inductive loop sensors. Each of these approaches has merits but also includes notable constraints and limits. Image Processing is a technique that utilizes cameras to get real-time photos or videos of traffic [10].

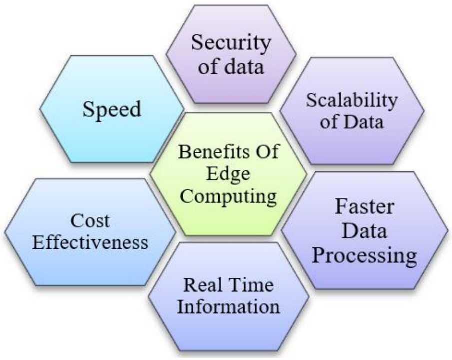

Figure 1: Edge computing benefits.

Advanced techniques like Reinforcement learning (RL), and (DRL), and hybrid approaches have shown considerable advances in traffic management, notably in terms of reduced trip times and emissions. However, scalability remains a critical issue that requires more work. Most studies have focused on single junctions. There is a considerable research gap in optimizing multiple intersections. Recent algorithms are a (Deep Q-learning Network) approaches that begin with no prior knowledge and exhibit significant unpredictability in action selection. This can result in poor control effects initially while this randomness lessens as the agent gets experience, it remains a drawback in the early phases of deployment, this can lead to inefficiencies in adapting to dynamic traffic environments, complexity in the implementation and tuning of the model [5]. In contrast, a generalized architecture of a modern adaptive traffic light control system is depicted in Figure 4.

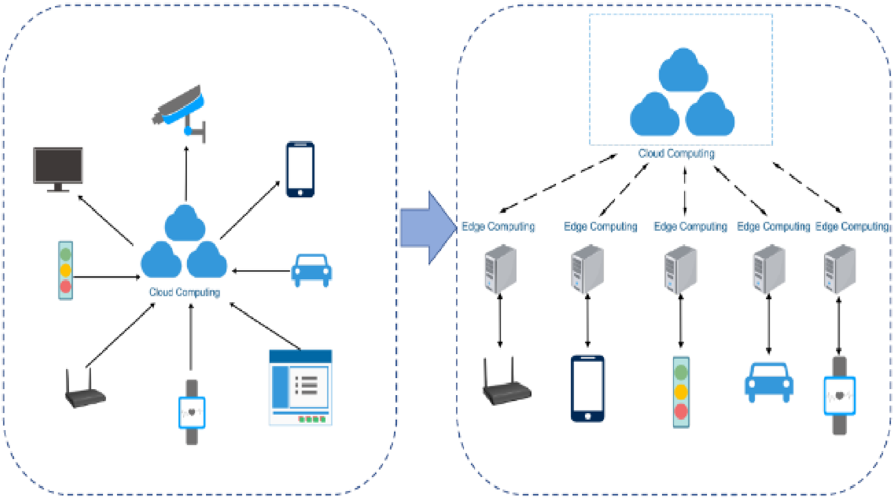

Figure 2: Cloud computing and edge computing [17].

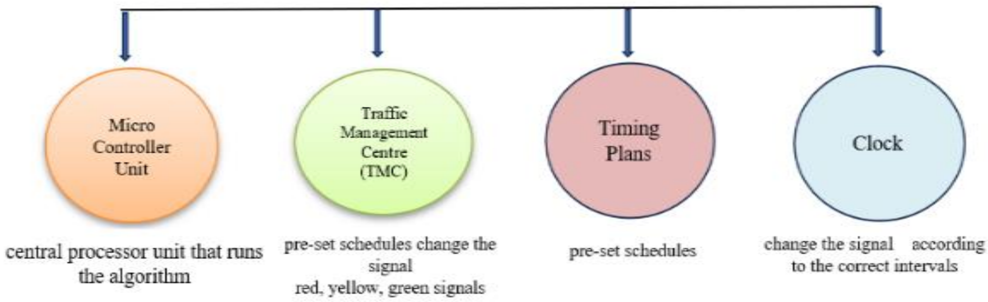

Figure 3: The basic components of the traditional traffic.

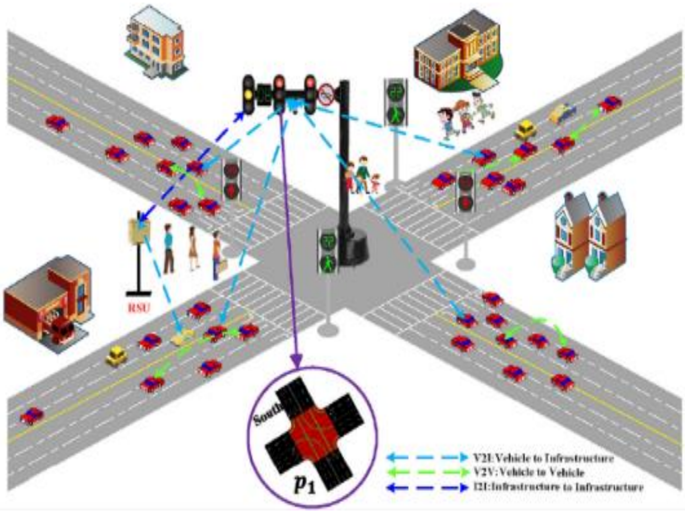

Figure 4: Adaptive traffic light control system [25].

This limitation affects the real time responsiveness needed for optimal traffic management [5]. With the congestion problem common in smart cities due to a rapidly growing number of vehicles, this study aims to solve several issues:

1) Control more than one intersection.
2) The algorithm's ability to adapt the green duration time according to the street density.
3) Integration of edge computing close to the sensors ensures accurate, real-time information based on specific road factors and parameters such as the density.
4) Constructing a system that can adapt to various road conditions.
5) This study can not only improve the efficiency of urban traffic and reduce traffic congestion, but also provide new ideas and methods for urban traffic management.
6) Constructing a unified, flexible, and quick-response traffic management system to update traffic lights in real-time with roadway modifications.
7) Achieves better scalability and reduce AWT on multi-intersections;
8) Introduce nonlinear algorithm (NMATCS) that employs an exponential function to control the traffic system.
9) Process data locally with edge computing and scale to multi-intersection networks with zero training.

The remainder of this paper is structured as follows: Section 2 reviews related work on intelligent traffic light control and the role of edge computing in IoT-based systems. Section 3 describes the proposed nonlinear system, detailing its architecture, mathematical model formulation with the justification, and simulation workflow. Section 4 presents the implementation environment, discusses the experimental results and performance comparisons with existing methods. Section 5 presents conclusion, outlines limitations and directions for future work.

2 RELATER WORK

A mechanism to control traffic congestion has become imperative due to the rise in both the number of people and vehicles, this section will cover several approaches that were developed to improving traffic management. Considering [11] fixed traffic signal (FTS) latencies cause approximately 10% of global traffic delays, when urbanization first began, many government organizations recognized the necessity of traffic control at (Rls) and implement old-age manual traffic signalized systems, in the manual traffic signal control (TSC) system indicated that over 295 million traffic hours are lost in developed nations due to (FTS) an intelligent transportation system (ITS) is the solution to handle the challenging problem of traffic control and management at (Ris), known as the adaptive traffic signal control (ATSC) system, which automatically adjusts the traffic durations times for each direction based on current traffic circumstances to improve traffic flow. These studies discovered that the traditional system has limitations, including the inability to share real-time traffic information with other nodes and its restriction to only nearby vehicles which is insufficient to manage traffic in a whole city [12], [13].The study addresses the drawbacks of conventional traffic signal control techniques, including their high purchase and installation costs, it highlights the necessity of more sophisticated techniques, such as reinforcement learning, to enhance adaptive traffic signal management systems and successfully handle traffic congestion [14]. The conventional methods are unable to provide a balanced distribution of traffic, which results in congestion [15]. Figure 3 illustrates traditional traffic control systems. The deployment of the smart traffic management system (STMS) is projected to drastically lower or reduce travel times by dynamically modulating traffic signal timings depending on current conditions of the road [16]. The study underlines the importance of adaptive solutions, especially those based on vehicular ad hoc network (VANET) technology, which improves traffic flow and lowers congestion at metropolitan crossings [8]. According to this study, traffic flow improved and congestion decreased when machine learning algorithms were integrated with Internet of Things sensors and linked cars for adaptive signal control and real-time traffic monitoring [16]. A comparative summary of key related studies is provided in Table 1.

Table 1: Summary of some published techniques for adaptive traffic light systems.

Adaptive Traffic Light System						
Ref.	Author-year	Description	Methodology	Tools	Focus area	Key finding
[16]	Jaleel et al., 2020	This study concentrates on reducing congestion using Edge computing and machine learning technologies to enhance the effectiveness of traffic control systems. As a Multi-agent Reinforcement Learning system, it considers traffic control as one of its agents acting to command through the learning optimal policy in signals.	-employing Deep Q-Network for value-function approximation and comparing its performance with Proximal Policy Optimization. -compares the developed system with centralized and decentralized traffic control systems.	-edge computer for computational acceleration -Deep Q-Networks, and Proximal Policy Optimization algorithms. -Traffic simulation is done using the SUMO traffic simulator, forming part of the Reinforcement Learning framework.	-traffic congestion mitigation through adaptive traffic control at signalized intersections, utilizing machine learning algorithms in a collaborative and decentralized manner.	-collaborative state-space formed by including neighboring signal phases results in performance comparable to centralized systems but at a lower cost. -The Deep Q-Network based system shows promising results in efficiently managing traffic flow and reducing average wait times at intersections.
[17]	Wu et al., 2020	This paper introduces a multi-agent auto communication algorithm (MAAC) for traffic light control. It deals with edge computing and MARL as optimal traffic signal schemes.	-using MAAC, by integrating MARL with an auto communication protocol inside the edge computing environment. -focusing on coordinated training among traffic signal agents to maximize control tactics.	- edge computing devices at traffic signals -reinforcement learning algorithms -CityFlow traffic simulator for testing and validating the algorithm.	- traffic signal control, specifically on improving traffic flow and reducing congestion at urban intersections using edge computing and MARL.	- leveraging edge computing, significantly improves traffic flow efficiency and reduces congestion compared to traditional methods.
[18]	Ajayi et al., 2020	The paper discusses a Smart Traffic Management System (STMS) using image processing to manage traffic at intersections.	-employing image detection using Single Shot Detector (SSD) for vehicle detection and a priority-based scheduling algorithm for traffic signal control.	-raspberry Pi for edge computing -SSD for image processing. -Java for the prototype software.	- reducing delays at road intersections through an intelligent traffic light system based on real-time vehicle detection.	-STMS significantly reduces service interruptions and delays compared to traditional traffic lights -improving traffic flow efficiency at intersections.
[19]	Dhingra et al., 2021	The paper presents an IoT-based framework combining fog and cloud computing for smart traffic monitoring. It focuses on congestion monitoring and traffic light management.	-integration of fog computing for real-time data processing - cloud computing for data storage and further analysis. - IoT devices and sensors for data collection.	-Intel Edison Kit for Arduino as the fog node -ultrasonic sensors for vehicle detection -ThingSpeak and Twitter for data storage and communication.	- efficient traffic monitoring - congestion detection - traffic light management using IoT technology within a fog-cloud computing architecture.	-this integrated fog and cloud computing system significantly improves response times and bandwidth efficiency compared to traditional cloud-only systems.

An adaptive traffic light system based on a random forest regressor model proved successful in controlling traffic lights at junctions as it produced a 30.8% decrease in traffic. Congestion [20]. The suggested system calculates traffic density along the route using real-time video and image processing techniques, therefore facilitating dynamic traffic control. Effective signal and number plate recognition follow from the extraction of sensor data and traffic density from cameras utilizing digital image processing technologies [20].The study shows a two-level optimization model that finds the best lengths of both the green light and traffic light cycles across a network. MATLAB was used to solve the created optimization problems, and the queue lengths that came up from the traditional deterministic optimization problem [21].

3 PROPOSED PROTOCOL

In this section, the development and validation of a (NMATCS) nonlinear adaptive traffic light control system are described. the system is intended to optimize the duration of green lights at intersections. Enabling dynamic signal adjustments to reduce vehicle AWT and ensuring no time is wasted on any of the traffic signals phase for clarity and coherence. The section is structured into four parts: proposed design architecture, the novel mathematical model formulation and assumptions, mathematical justification of the main nonlinear equation, algorithm's input parameters and flowchart.

3.1 Proposed Model Architecture

The proposed model is composed of edge devices deployed near intersections, integrated with IoT sensors such as inductive loops, infrared counters, and cameras (Fig. 5). These sensors capture real-time vehicle count data, which is pre-processed locally at the peripheral edge to eliminate delays and decrease dependency on centralised cloud services. Each intersection includes four roads (North, South, East, and West), all of which are monitored for vehicle density. The edge node runs the control algorithm, which is composed of a microcontroller or mini-computer that is determining the green and red light durations for each direction. This decentralised design improves scalability and fault tolerance, enabling responsive adaptation even in high-traffic scenarios.

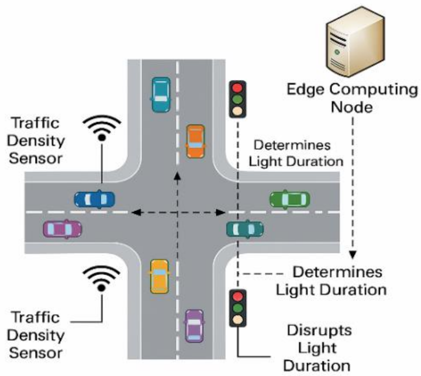

Figure 5: Proposed model architecture.

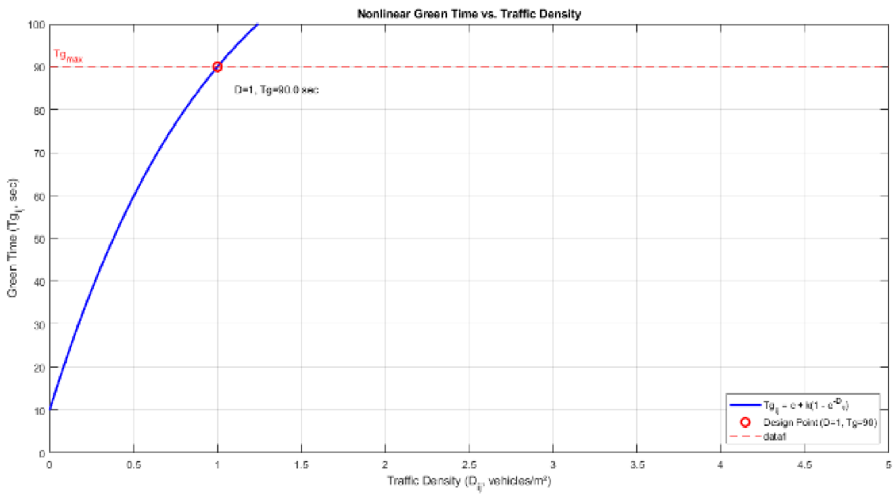

Figure 6: Nonlinear response of green time to traffic density in the NMATCS model (T_{gij} vs. D_{ij}).

3.2 Mathematical Model Formulation and Assumptions

The core of the algorithm is a nonlinear equation that regulates and adjusts the green and red traffic light durations:

- The algorithm's fundamental component is a nonlinear equation that regulates and adjusting the duration times of green and red traffic light signal is (1). Where T_{gij} is the green duration time (sec) to intersection j for road i, and c is the base green duration time for min density, c = T_{gij} (min), while c is a coefficient representing the base green duration time for min density of the road.

$$T_{gij} = c + k\left(1 - \frac{1}{\exp(Dij)}\right). \quad (1)$$

k is a coefficient represents how much the green duration increases per additional vehicle density can be calculated using (2)

$$k = (T_g \text{ Max} - T_g Min) / 0.632. \quad (2)$$

- The AWT sec for all N vehicles it is the specific delay associated with the intersection can be determined by subtracting this ideal traversal time from the total time spent in the network. can be calculated using (3), [22]. Where N = total number of vehicles in the simulation. W_i= waiting time of the vehicle, defined as:

$$W_i = t - t_o(i)$$

$$W_{avg} = \frac{1}{N}\sum_{i=1}^{N} W_i. \quad (3)$$

- Red duration time for road i, red time equals the sum of green times for conflicting directions in the same cycle, is done by summing the green duration times for the three directions, except for the specific direction can be calculated using (4), [23]

$$T_{rij} = \sum_{\substack{r=1 \\ r\neq i}}^{4} T_{gij} \quad . \quad (4)$$

- Total cycle time required to complete a full cycle of green phases in all directions at a one intersection. can be calculated using (5),

$$T_{cycle} = \sum_{i=1}^{N} T_g^{\,i} \rightarrow = (Tgn + Tgw + Tgs + TgE). \quad (5)$$

3.3 Mathematical Justification of the Main Nonlinear Equation

To calculate the green duration time (sec) [T_{gij} = c + k. (1- $\frac{1}{\exp(Dij)}$)] was obtained by trial and error because we are dealing with a parameter that changes with time, which is the street density (D_{ij}).

As it is not possible to predict its increase or decrease in every fraction of a second, this parameter was multiplied by the exponential function minus the maximum value of the street density, the exponential function exp (D_{ij}) ensures non-linearity.

The effect of D_{ij} on T_{gij} is not proportional, reflecting real phenomena were large Densities or dissimilarities have diminishing impacts. And ensures boundedness.

The term [$\frac{1}{\exp(Dij)}$] ensures T_{gij} cannot exceed c + k, preventing unrealistic values, so, the exponential ensures bounded growth and prevents the occurrence of excessive green periods due to minor density fluctuations making it suitable for systems where effects diminish with increasing D_{ij} while the maximum value guarantee T_{gij} has a maximum value of c + k when D_{ij} is large, and when D_{ij} is zero, it's c + k. (1- $\frac{1}{1}$) = c, if D_{ij} is zero, exp (0) is 1, so (1- $\frac{1}{1}$) = 0 then T_{gij} would be c. so, the equation models a value that starts at c when D_{ij} is zero and asymptotically approaches c + k as D_{ij} increases.

That's a common logistic-type growth but inverted because it's 1 minus the inverse exponential. Figure 6. of the green duration time T_{gij} versus traffic density D_{ij} illustrating the nonlinear relationship, the curve goes up strongly at low values and flattens out as D_{ij} goes up, which shows that the benefits are decreasing.

3.4 Algorithm's Input Parameters and Flowchart

The model incorporates configurable green time bounds to allow flexibility in adapting to varying traffic conditions.

Configurable green time bounds:

- (Tg) Max = 90 sec (default, user-adjustable);
- (Tg) Min = 10 sec (default, user-adjustable);
- Real-time density metric (D_{ij}) (vehicles/m²) that is derived from vehicle counts can be calculated using (6), where A_0 is the area per one vehicle, M is number of vehicles, and A is the area of the road. $A = (L_i * W_i)$, where L_i the length of the road, W_i is the road width:

$$D_{ij} = \frac{M * A_0(m^2)}{A(m^2)} \qquad 0 \le D_i \le 1. \quad (6)$$

4 SIMULATION RRESULTS AND ANALYSIS

The simulations were performed using the proposed algorithm (NMATCS). The simulation process utilised real data collected from intersections in Hangzhou, China, which was stored in structured JSON files and parsed using MATLAB R2024a for four different scenarios at four intersections. Data was collected for each intersection, including four directions (North, West, South, and East) over a one-hour period. All simulation scenarios were executed over 100 iterations; each iteration can be seen as a discrete snapshot or time segment to ensure statistical reliability and account for the dynamic nature of vehicle arrivals.

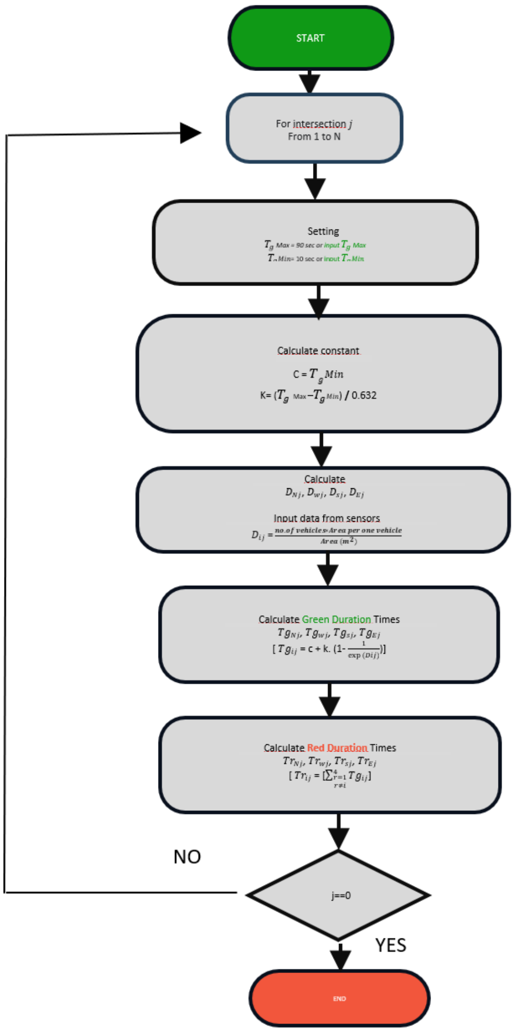

Figure 7: Proposed Algorithm's flowchart.

The section is structured into four parts: simulation parameters and assumptions, evaluation metrics, results of scenarios, and comparison of AWT and the performance with different algorithms. The whole structure of the simulation is shown in the following Figure 7.

4.1 Simulation Parameters and Assumptions

The simulation was performed using a real data set, with specific input parameters and assumptions. Table 2 explains the details.

Table 2: Real road statistic parameters and assumptions.

Parameter	Value/Range
Max green time (T_{gij})	90 seconds
Min green time (T_{gij})	10 seconds
Simulation iterations per intersection	100
Road length (L)	12 meters
Road width (W)	9 meters
Vehicle area (A_0)	6 m^2
Max vehicles per road	18
Min Density per one vehicle (D_{min})	0.055
Max density for full road (D_{Max})	1.0
Number of vehicles (M) for intersection 1	1848
Number of vehicles (M) for intersection 2	1375
Number of vehicles (M) for intersection 3	1565
Number of vehicles (M) for intersection 4	1761

4.2 Evaluation Metrics

Several important metrics were used to evaluate traffic algorithm's performance such Green Duration time (T_{gij}) and AWT [24] and Total Cycle Time (T_{cycle}). Their equations were previously mentioned.

4.3 Results of Scenarios

This section presents the outcomes of four distinct simulation scenarios designed to evaluate the performance of the proposed traffic signal control algorithm under varying traffic conditions. Each scenario focuses on a different intersection configuration and traffic distribution pattern, highlighting how the algorithm adapts green light durations based on real-time vehicle density measurements. For each case, the results include both the average vehicle densities and the corresponding average green durations for all roads at the intersection, supported by detailed tables and visualized through dual y-axis plots.

1) 1st Simulator, Intersection 1 displays a traffic arrangement with a skewed vehicle distribution, where Road 2 (West) has a significantly higher average vehicle density (0.413 veh/m) compared to the other roads (0.214-0.242 veh/m). This is a realistic scenario in which one direction experiences a heavier traffic inflow during the simulation window. Table 3. below illustrates the results of the average densities and average green duration times for each road at Intersection 1. Figure 8. The dual y-axis plot displays the results of allocating the green light duration time by applying the proposed algorithm based on the density value of each road direction.
2) 2nd Simulation Scenario. Figure 9 and Table 4 illustrate the results of the average densities and average green duration times for each road at Intersection 2.
3) 3rd Simulation Scenario. Figure 10, and Table 5. illustrates the results of the average densities and average green duration times for each road at Intersection 3.
4) 4th Simulation Scenario. Figure 11 and Table 6. illustrate the results of the average densities and average green duration times for each road at Intersection 4;

4.4 Comparison Results on Four Intersections

Table 7 displays the outcomes of four separate junctions' simulations were examined and contrasted to see how well the suggested adaptive traffic signal control method performed. The total cycle time and average vehicle waiting time are key performance measures that are influenced by traffic density and flow patterns at each intersection.

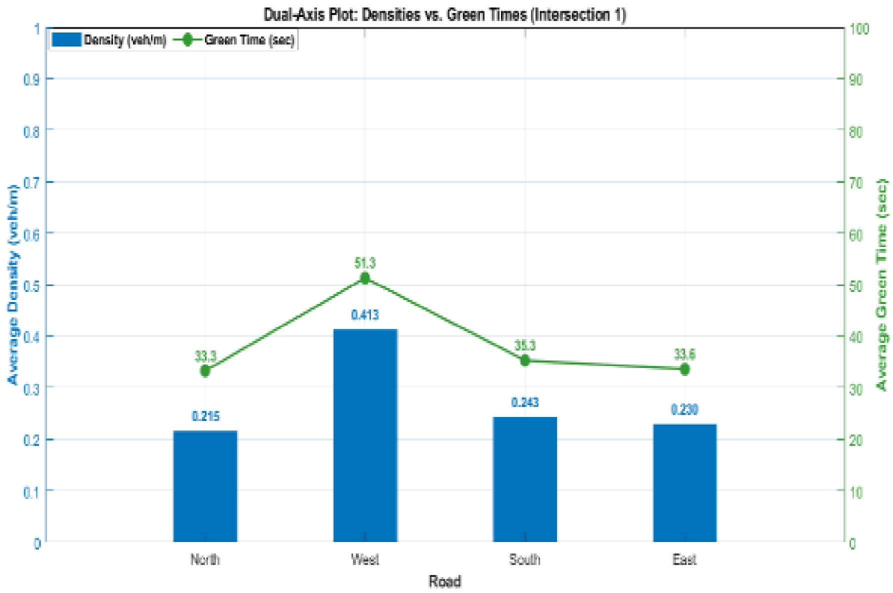

Figure 8: Density vs. Green Time – Intersection 1.

Table 3: Intersection 1 Results Summary.

Road	Average Density (veh/m)	Average Green Time (sec)	Average Red Time (sec)
North	0.2153	33.3	122.4
West	0.4133	51.3	99.3
South	0.2431	35.3	119.3
East	0.2295	33.6	122.5

Table 4: Intersection 2 Results Summary line so it is centered.

Road	Average Density (veh/m)	Average Green Time (sec)	Average Red Time (sec)
North	0.183	30.1	93.82
West	0.12	23.86	100.06
South	0.181	29.76	94.16
East	0.28	40.2	83.72

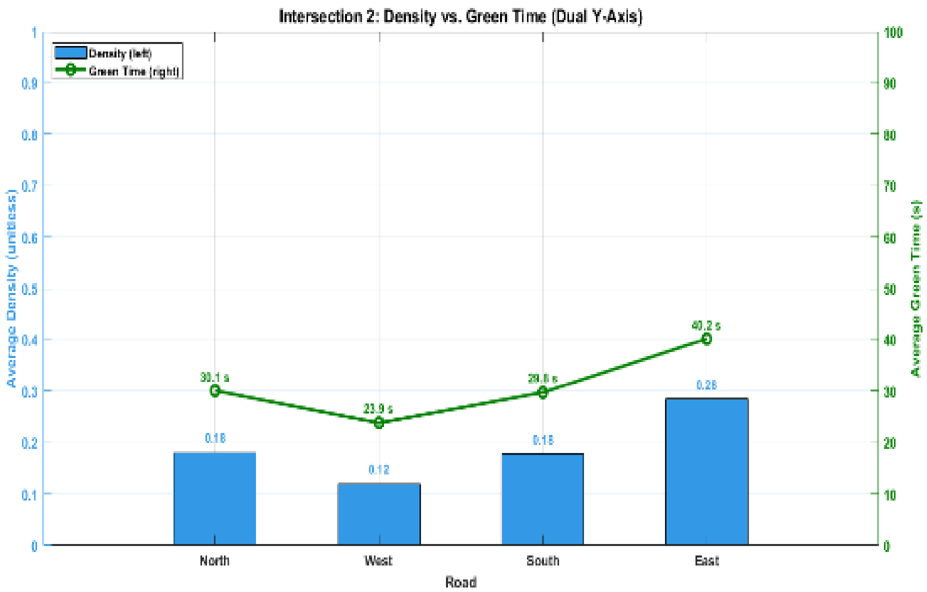

Figure 9: Density vs. Green Time – Intersection 2.

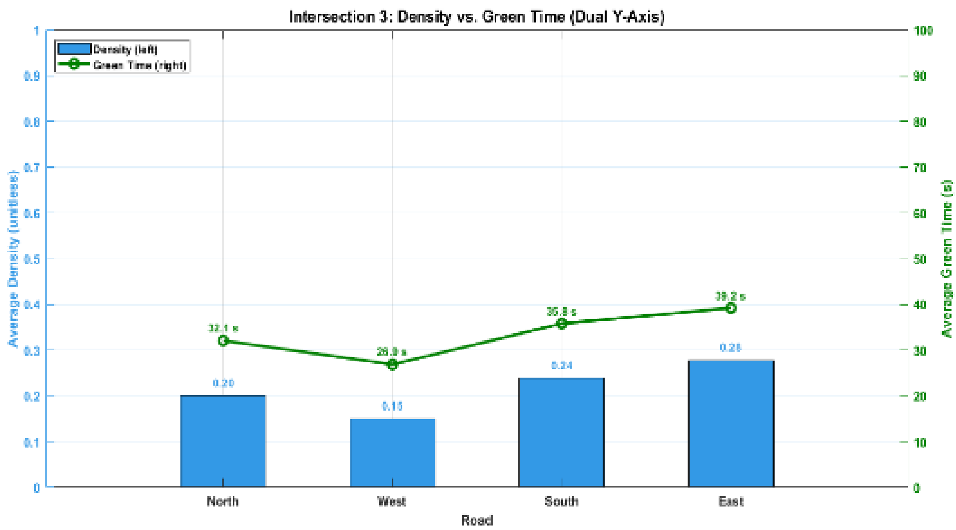

Figure 10: Density vs. Green Time – Intersection 3.

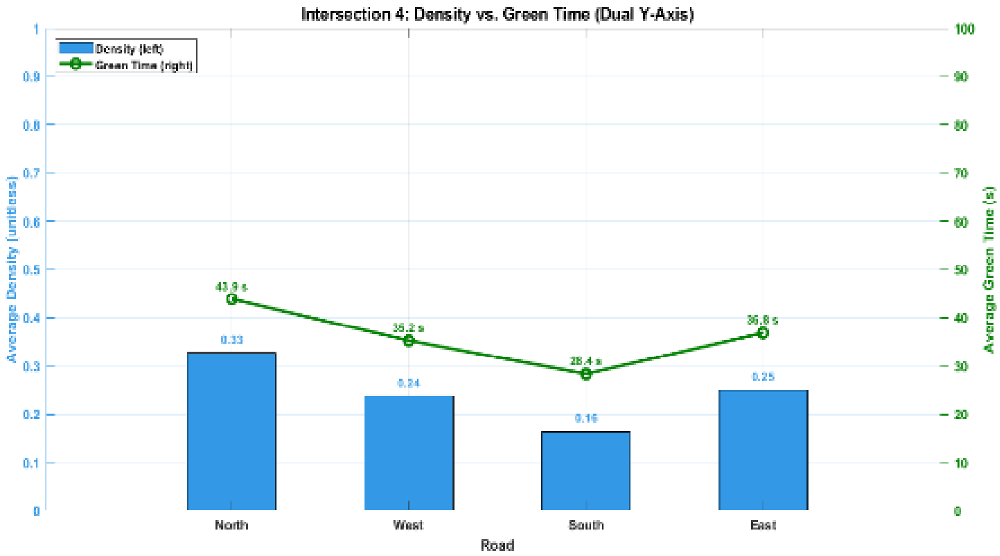

Figure 11: Density vs. Green Time – Intersection 4.

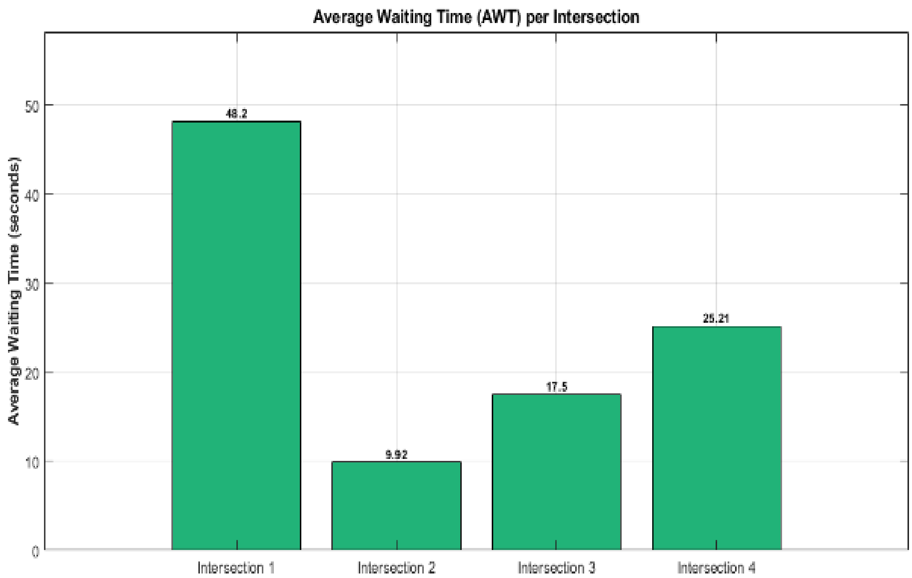

Figure 12: Comparison of Average Waiting for four intersections.

Table 5: Intersection 3 Results Summary line so it is centered.

Road	Average Density (veh/m)	Average Green Time (sec)	Average Red Time (sec)
North	0.20	32.1	101.9
West	0.15	26.89	107.1
South	0.24	35.82	98.2
East	0.28	39.21	94.8

Table 6: Intersection 4 Results Summary line so it is centered.

Road	Average Density (veh/m)	Average Green Time (sec)	Average Red Time (sec)
North	0.33	43.85	100.42
West	0.24	35.24	109.03
South	0.16	28.36	115.91
East	0.25	36.82	107.45

Table 7: Comparison of AWT and Cycle Times for per intersection.

Intersection	Average waiting time (sec)	Average total cycle time (sec)	Key Insight
Intersection 1	48.2 sec	153.48 sec	High congestion in one direction; others low
Intersection 2	9.92 sec	123.92 sec	Balanced traffic across all roads
Intersection 3	17.5 sec	134.03 sec	Moderate mixed traffic densities
Intersection 4	25.21 sec	144.27 sec	Heavy traffic in all directions

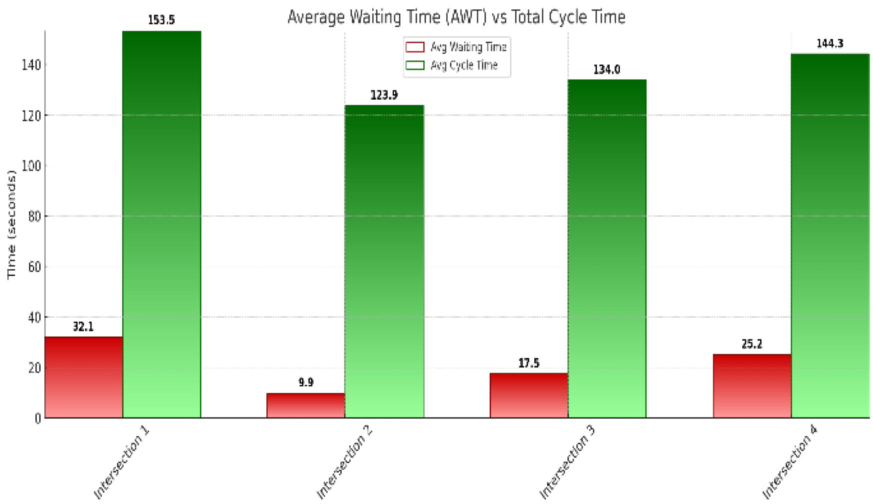

Figure 13: Comparison of AWT and Cycle Times for per intersection.

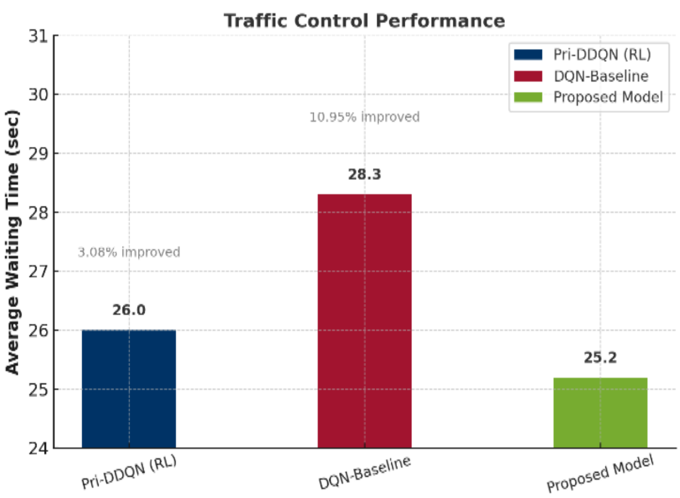

Figure 14: Comparison with different algorithms.

Intersection 1 was the most congested, with an AWT of 48.2 seconds. Figure 12. indicates that vehicles at this intersection encountered severe delays,Intersection 2 had the least traffic congestion, with an AWT of 9.92 seconds and the shortest average cycle time of 123.92 seconds. Intersection 3 maintained a moderate balance, with an AWT of 17.50 seconds and a cycle time of 134.03 seconds, proving the algorithm's capacity to adjust appropriately to changing densities. Meanwhile, Intersection 4 had a longer cycle time (144.27 seconds) and a greater waiting time (25.21 seconds), indicating higher density in all directions but reasonably even signal dispersion. Intersection 2 demonstrates how well the adaptive system handled balanced traffic circumstances.

4.4 Compares Average Waiting Times and the Performance with Different Algorithms

The average waiting time was compared among Pri-DDQN, DQN-Baseline [25] and the proposed nonlinear model (NMATCS). The results show an improvement of 3.08% over Pri-DDQN and 10.95% over the DQN baseline. Figures 13 and 14 show that the DQN-Baseline has the greatest (AWT).

5 CONCLUSIONS

This paper presents the results of satisfying the key research objectives and addressing the issue area effectively. The following conclusions were drawn from the study and its results:

1) Traffic congestion in urban areas is a significant issue as the global road infrastructure struggles to keep up with the increasing demand for mobility.
2) A novel adaptive nonlinear traffic control equation T_{gij} = c + k. (1- $\frac{1}{\exp(Dij)}$) that adapts the traffic green duration time based on the road vehicles density in real that adapts the traffic green duration time based on the road vehicles density in real time without requiring historical data or learning (zero training).
3) A MATLAB-based simulation framework for evaluating and testing adaptive signal control strategies using both real and synthetic data.
4) The model was validated and deployed across various intersections, demonstrating its ability to manage diverse traffic flows and effectively scale.
5) The suggested model had the lowest AWT of 25.2 seconds, compared to 26.0 seconds for Pri-DDQN (2024) and 28.3 seconds for DQN-Baseline.
6) Our algorithm achieved an improvement rate of up to 3.08% when compared to Pri-DDQN and 10.95% when compared to DQN-Baseline.
7) The new proposed nonlinear equations and Integration of edge computing close to the sensors ensures accurate, real-time information based on specific road factors.
8) The computational efficiency was significantly lower because our model was not requiring prior learning or training.
9) We note through the experiments that there was no wasting time for the signal more than the actual time required, as the algorithm adapts systematically to the value of the road density.

6 LIMITATIONS AND RECOMMENDATIONS FOR FUTURE WORK

While the proposed traffic signal control model demonstrates strong adaptability to varying traffic densities, several limitations remain that present opportunities for further enhancement. Addressing these constraints can improve system accuracy, robustness, and applicability in diverse real-world settings. The following recommendations focus on integrating additional detection capabilities, advancing the decision-making framework, enhancing sensor reliability under adverse conditions, and expanding the scope of testing to encompass a broader range of urban traffic configurations:

1) Integrate detection mechanisms for emergency vehicles, fire engines, and pedestrian crossing signals.
2) Developing a hybrid model that combines rule-based decision-making with reinforcement learning for improved long-term adaptation.
3) The model assumes continuous traffic flow without accounting for weather conditions (e.g., rain, fog, sudden sensor failures) affecting sensor reliability. Therefore, we recommend using alternative sensors, such as the Kalman Filter. This filter predicts the next expected measurement based on past sensor values. If a new reading significantly deviates from the expected values (indicating it might be an outlier), the Kalman Filter will either reduce its influence or reject it altogether.

4) Testing the system in various urban environments such as grid layouts and roundabouts to demonstrate increased scalability.

REFERENCES

[1] P. Raja, S. S. Kumar, D. Yadav, and T. Singh, "The Internet of Things (IoT): A review of concepts, technologies, and applications," International Journal of Information Technology and Computer Engineering, no. 32, pp. 21–32, 2023, doi: 10.55529/ijitc.32.21.32.

[2] A. Gaur, B. Scotney, G. Parr, and S. McClean, "Smart city architecture and its applications based on IoT," Procedia Computer Science, vol. 52, pp. 1089–1094, 2015, doi: 10.1016/j.procs.2015.05.122.

[3] H. Omar Al-Sakran, "Intelligent traffic information system based on integration of Internet of Things and agent technology," 2015. [Online]. Available: www.ijacsa.thesai.org.

[4] F. Al-Turjman and A. Malekloo, "Smart parking in IoT-enabled cities: A survey," Sustainable Cities and Society, vol. 49, p. 101608, Aug. 2019, doi: 10.1016/j.scs.2019.101608.

[5] Proc. 2020 IEEE Int. Conf. Internet Things (iThings), Green Comput. Commun. (GreenCom), Cyber, Phys. Social Comput. (CPSCom), Smart Data (SmartData), Congr. Cybermatics (Cybermatics), Beijing, China, 2020, doi: 10.1109/iThings-GreenCom-CPSCom-SmartData-Cybermatics50389.2020.00012.

[6] S. S. Siripuram, "Leveraging edge computing for scalable real-time cloud systems," [Online]. Available: www.ijfmr.com.

[7] L. Liu, C. Chen, Q. Pei, S. Maharjan, and Y. Zhang, "Vehicular edge computing and networking: A survey," arXiv: Signal Processing, 2019. [Online]. Available: https://arxiv.org/pdf/1908.06849.pdf.

[8] R. Kolapo, F. M. Kawu, A. D. Abdulmalik, U. A. Edem, M. Young, and E. C. Mordi, "Edge computing: Revolutionizing data processing for IoT applications," International Journal of Science and Research Archive, vol. 13, no. 2, pp. 23–29, 2024, doi: 10.30574/ijsra.2024.13.2.2082.

[9] A. Kamput, C. Dechsupa, W. Vatanawood, and S. Pomsiri, "Scalable timed-automata models for traffic light control systems: Challenges and solutions in formal verification," IEEE Access, vol. 12, pp. 124260–124281, 2024, doi: 10.1109/ACCESS.2024.3455097.

[10] J. Zhang, H. Guo, J. Liu, and Y. Zhang, "Task offloading in vehicular edge computing networks: A load-balancing solution," IEEE Transactions on Vehicular Technology, vol. 69, no. 2, pp. 2092–2104, 2020, doi: 10.1109/TVT.2019.2959410.

[11] M. Noaeen et al., "Reinforcement learning in urban network traffic signal control: A systematic literature review," Expert Systems with Applications, vol. 199, p. 116830, 2022, doi: 10.1016/j.eswa.2022.116830.

[12] D. Bhattacharyya, T. H. Kim, and S. Pal, "A comparative study of wireless sensor networks and their routing protocols," Dec. 2010, doi: 10.3390/s101210506.

[13] M. S. Farooq and S. Kanwal, "Traffic road congestion system using the Internet of Vehicles (IoV)," 2017.

[14] S. Damadam, M. Zourbakhsh, R. Javidan, and A. Faroughi, "An intelligent IoT-based traffic light management system: Deep reinforcement learning," Smart Cities, vol. 5, no. 4, pp. 1293–1311, 2022, doi: 10.3390/smartcities5040066.

[15] M. Q. Kheder and A. A. Mohammed, "Real-time traffic monitoring system using IoT-aided robotics and deep learning techniques," Kuwait Journal of Science, vol. 51, no. 1, Jan. 2024, doi: 10.1016/j.kjs.2023.10.017.

[16] A. Jaleel, M. A. Hassan, T. Mahmood, M. U. Ghani, and A. Ur Rehman, "Reducing congestion in an intelligent traffic system with collaborative and adaptive signaling on the edge," IEEE Access, vol. 8, pp. 205396–205410, 2020, doi: 10.1109/ACCESS.2020.3037348.

[17] Q. Wu, J. Wu, J. Shen, B. Yong, and Q. Zhou, "An edge-based multi-agent auto communication method for traffic light control," Sensors, vol. 20, no. 15, pp. 1–16, Aug. 2020, doi: 10.3390/s20154291.

[18] O. A. Ajayi, I. O. Bagula, and N. A. Olasupo, "Effective management of delays at road intersections using smart traffic light system," in e-Infrastructure and e-Services for Developing Countries, R. Zitouni and P. M. H. S. Agueh, Eds. Cham, Switzerland: Springer, 2020, pp. 84–103.

[19] S. Dhingra, R. B. Madda, R. Patan, P. Jiao, K. Barri, and A. H. Alavi, "Internet of Things-based fog and cloud computing technology for smart traffic monitoring," Internet of Things, vol. 14, Jun. 2021, doi: 10.1016/j.iot.2020.100175.

[20] M. G. Raj, "Implementation on IoT-based traffic management system for emergency vehicles," International Journal for Science Technology and Engineering, vol. 11, no. 5, pp. 837–845, 2023, doi: 10.22214/ijraset.2023.51614.

[21] K. Stoilova and T. Stoilov, "Optimization models for urban traffic management," WSEAS Transactions on Systems and Control, vol. 18, pp. 187–194, 2023, doi: 10.37394/23203.2023.18.19.

[22] Transportation Research Board, Highway Capacity Manual, 6th ed. Washington, DC, USA: The National Academies Press, 2016, doi: 10.17226/24798.

[23] R. Roess and W. McShane, "Traffic engineering," in Traffic Engineering, 4th ed. Pearson, 2010, ch. 12.

[24] P. Thakre, P. Bhalerao, A. Dongre, L. Bendey, I. Jaiswal, and C. Anikhindi, "Design and implementation of a dynamic traffic signal system with digital circuit and IoT integration for efficient traffic management," pp. 1–7, 2023, doi: 10.1109/ICCCNT56998.2023.10306612.

[25] Y. Zheng, J. Luo, H. Gao, Y. Zhou, and K. Li, "Pri-DDQN: Learning adaptive traffic signal control strategy through a hybrid agent," Complex and Intelligent Systems, vol. 11, no. 1, Jan. 2025, doi: 10.1007/s40747-024-01651-5.