

Design and Implementation of Intelligent Traffic Control Systems with Vehicular Ad Hoc Networks

Osita Miracle Nwakeze*

Department of Computer Science, Chukwuemeka Odumegwu Ojukwu University, Uli, Anambra State

E-mail: ma.nwakeze@coou.edu.ng

ORCID iD: <https://orcid.org/000-0002-3717-8182>

*Correspondence Author

Christopher Odeh

Department of Computer science, Dennis Osadebay University Asaba, Delta State, Nigeria

E-mail: odehchris100@gmail.com

ORCID ID: <https://orcid.org/0000-0002-4502-2970>

Obaze Caleb Akachukwu

Department of Computer science, Dennis Osadebay University Asaba, Delta State, Nigeria

E-mail: caleb.obaze@dou.edu.ng

ORCID iD: <https://orcid.org/0009-0005-9393-4113>

Received: 02 November, 2025; Revised: 11 December, 2025; Accepted: 20 January, 2026; Published: 08 February, 2026

Abstract: Urban traffic congestion can be considered as a significant problem, and it contributes to long travel periods, fuel usage, and environmental influence. This paper introduces an Intelligent Traffic Control System (ITCS) that consists of Vehicular Ad Hoc Networks (VANETs) and Reinforcement Learning (RL) to optimise the control of traffic signals. The system facilitates real-time two-way communication between vehicles and roadside units, which means that an RL agent can control signal phases adaptively according to the traffic metrics like the average delay, the queue length, and traffic throughput. The Kaggle VANET Malicious Node Dataset was used to simulate malicious or unreliable nodes and test the robustness of the systems. The RL agent has been trained on the SUMO simulator trained on TraCI through various episodes and learns to take actions that increase traffic movement with a minimum amount of congestion. The results of training are progressive, as cumulative rewards grow, and average delays and queue length reduce with epochs. Performance evaluation of the ITCS under peak-hour, off-peak, incident, and malicious node scenarios demonstrated substantial gains over conventional fixed-time controllers, with average delays reduced by 48–55%, queue lengths by 49–57%, and throughput increased by 28–35%. These results indicate the success of the blend of reinforcement learning with VANET-supported traffic control, which is an adaptive, data-driven, and robust solution to an urban intersection. Not only the RL-based ITCS enhances traffic flow and congestion, but is also resistant to communication anomalies, which indicates its scalability to be deployed in the current smart city traffic management.

Index Terms: Intelligent Traffic Control System (ITCS); Reinforcement Learning; Vehicular Ad Hoc Networks (VANETs); Traffic Signal Optimization; Urban Traffic Management

1. Introduction

The challenge of traffic congestion, delays, road accidents, and pollution of the environment have been among the continuous issues due to urbanisation and rapid population growth, which requires an efficient transport system that is equally responsive [1,2]. The conventional traffic control systems, usually anchored to fixed-time schedules or basic actuated controllers, cannot be flexibly adapted to a changing traffic environment [3,4]. With the further growth of vehicles population, the ineffectiveness of traditional systems becomes more noticeable, and it is time to find innovative solutions that could allow the dynamic regulation of traffic streams, improving safety and sustainability at the same time [5].

The recent technological developments in the field of vehicular communication, especially, Vehicular Ad-hoc Networks (VANETs), allow new possibilities of changing the traffic organisation. VANETs facilitate both Vehicle-To-Vehicle (V2V) and Vehicle-To-Infrastructure (V2I) communication between vehicles and roadside units to exchange real-time information about the position, the speed, and the intended manoeuvre [6]. This information sharing forms a basis of ITCS capable of forecasting and responding to changing traffic trends, congestion reduction, minimising emissions, and enhancing overall road network efficiency [7].

Smart traffic control systems, which rely on VANETs and are characterised by the use of modern communication protocols and modern computational intelligence methods such as optimization models and machine learning algorithms, are integrated [1,8]. Using the real-time vehicular data, these systems can adapt the signal timing in real-time, coordinate various intersections, and advise drivers on the best way to move along the routes and speed [9]. Moreover, edge computing combined with cloud-based coordination contributes to the scalability and strength of these systems, allowing them to be deployed on the city road network in large amounts and retain the low-latency decision-making at the intersection [2].

The paper is devoted to the design and implementation of a VANET-based ITCS with a particular emphasis on adaptive and learning-based control approaches to solving the complex traffic situation. The study addresses the design, communication stack, control algorithms, and simulation frameworks and assessment measures required to build the system and test it. This work will help in improving the creation of future transportation systems that can offer safer, smarter and sustainable mobility solutions by filling the gap between the emerging technology of vehicular networking systems and the latest traffic control methods.

2. Methodology

The research uses the Design Science Research (DSR) technique and concentrates on design and development phases, demonstration and evaluation phases, and communication phases. During the design and development stage, an Intelligent Traffic Control System (ITCS) based on Vehicular Ad Hoc Networks (VANETs) is simulated to allow real-time communication between vehicles and roadside units. The decision-making process based on reinforcement Learning is implemented into the control system to dynamically optimise the timing of traffic signals to traffic changes. It is demonstrated with the help of the open-source Simulation of Urban MObility (SUMO) platform along with the Traffic Control Interface (TraCI) providing the ability to interact with traffic lights, vehicles, and network components using Python in order to realistically test adaptive signal control strategies.

The system is tested with the simulation of varied traffic conditions such as peak, off-peak and congestion conditions and comparing intelligent system with the fixed-time controllers and actuated controllers. The three major measures of performance include average delay (the average waiting time of vehicles), length of the queues (vehicles which have been accumulated at intersections and therefore are indicators of congestion), and the throughput (vehicles passing through the network in a unit time). These metrics are directly obtained in the simulations of SUMO through TraCI and discussed statistically to prove the effectiveness of the system. Communication is the last step of the methodology in which the design, experimental setup, results, and contributions are organised in a system that guarantees reproducibility as well as showing practical relevance of the proposed ITCS.

2.1 Data Collection

The dataset of this paper is the VANET Malicious Node Dataset, which was obtained on Kaggle. This data set gives the communication logs in VANETs with the normal and malicious behaviour of nodes. It contains the packet delivery ratios, message transmission rate, end to end delays and attack indicators besides the corresponding class labels of either the node is acting in good faith or maliciously. The choice of the dataset is due to the fact that it offers a representative foundation to investigate ITCS that will work reliably even during the occurrence of security threats.

The information was downloaded off Kaggle and directly used as the main input towards the training and evaluation of the Reinforcement Learning (RL) model. With the help of this dataset, the research avoids the manual data collection in the real car settings, yet it achieves the realistic nature of communication and attack patterns within VANETs settings. The labelled characteristics of the dataset render it to be appropriate in supervised preprocessing and embedding it into an RL framework, where the system would learn to optimise the use of traffic in the face of the existence of malicious or abnormal communication behaviours.

While the Kaggle VANET Malicious Node Dataset is labelled for node behaviour classification, its primary utility in this study is to provide realistic patterns of communication anomalies (packet loss, delay spikes, message falsification). These patterns are extracted and mapped onto the simulated VANET within SUMO to stress-test the ITCS's resilience. The control objective remains traffic optimization; the dataset serves to simulate a challenging communication environment, not to train a node classifier.

2.2 Data Preprocessing

The dataset used in this paper is the VANET Malicious Node Dataset sourced out of Kaggle, which comprises of labelled vehicular communication packets, that was meant to record normal and malicious node behaviour in VANETs. Given the fact that the dataset is a raw communication traces, preprocessing was necessary to prepare it to be used in

training the RL model. Data cleaning was the starting point, as it was necessary to determine the presence of missing and repetitive entries, delete them to guarantee the quality of training samples [10]. One-hot encoding was used to convert categorical variables (node roles or attack types) into numerical values [11], whereas continuous features (the rate of packet transmissions, end to end delay, and delivery ratio of packets) were normalized using min-max scaling to ensure the similarity of features [12]. Random under-sampling and Synthetic costly oversampling with the Synthetic Minority Over-sampling Technique (SMOTE) were used to provide the balance between normal and malicious records based on classes and provide no bias in the model towards the majority [13]. Lastly, feature selection was used to obtain the most meaningful variables that affect the process of malicious node detection to minimise noise and dimensionality in the input space [10]. The output of this preprocessing pipeline was a clean and balanced dataset, which was then divided into training, validation, and testing sets, which served as the basis of developing and testing the proposed RL-based ITCS.

The applied preprocessing steps (SMOTE, feature selection) were used solely to create a balanced and representative sample of communication scenarios from the Kaggle dataset. This processed dataset is then used to stochastically trigger anomaly events within the simulation environment. The RL agent itself does not perform classification; it learns a control policy from traffic states that may be corrupted by these pre-defined anomaly patterns.

3. Modelling of an Intelligent Traffic Control System (ITCS) Leveraging Vehicular Ad Hoc Networks (VANETS)

ITCS are intended to manage traffic, minimise congestion, and enhance safety on the roadway based on real-time data gathered at the vehicle and the infrastructure level. With VANETS, ITCS is capable of using Vehicle-To-Vehicle (V2V) and Vehicle-To-Infrastructure (V2I) communications to dynamically track the traffic condition and better organise traffic lights than traditional fixed-time controllers. This will enable intersections and corridors to react to the changing traffic needs, accidents, or abnormal traffic patterns on the fly. An ITCS that is based on VANET comprises of three fundamental conceptual elements as depicted in Fig. 1. First, the vehicular environment which is the application of vehicles as mobile nodes, capable of sharing information including speed, position and the intended routes. Second, the

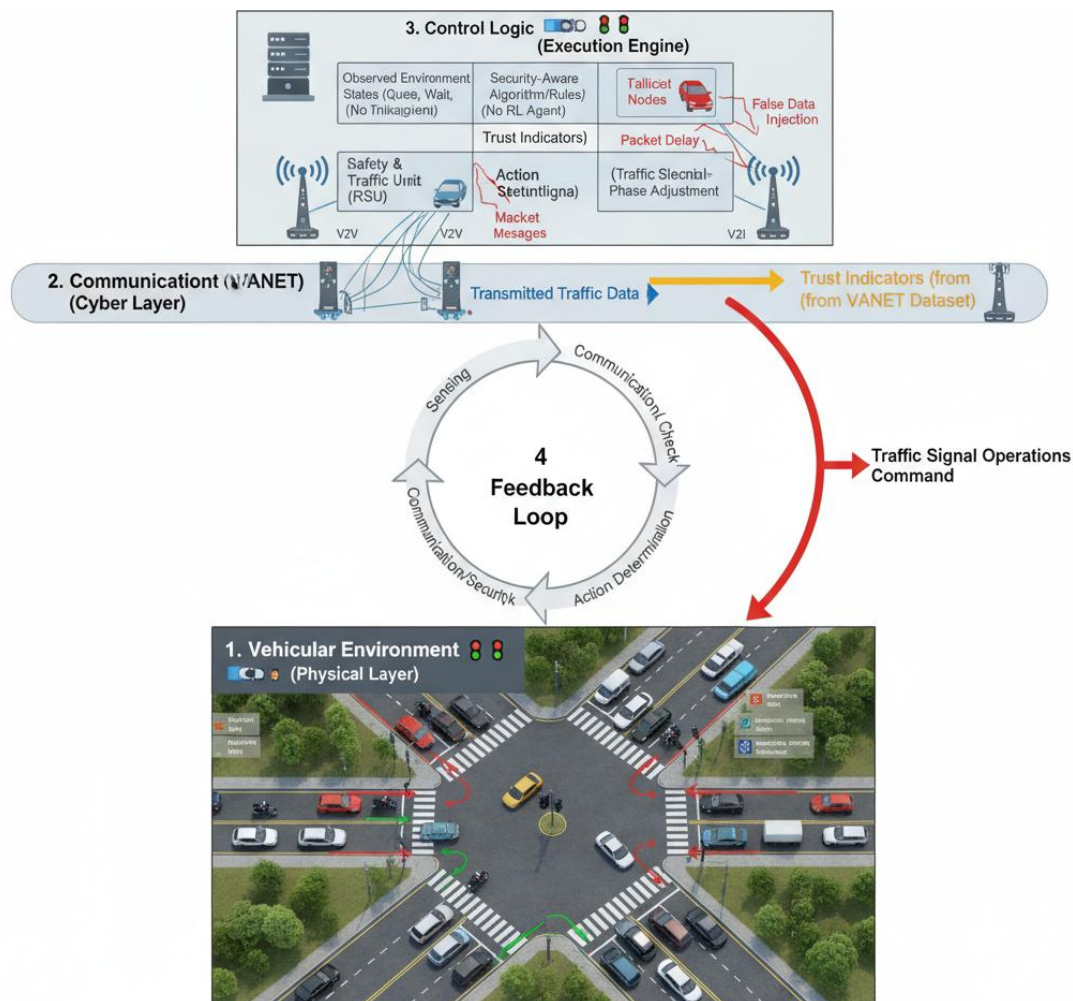


Fig. 1. Structure of ITCS Framework based on VANET

communication network that can also facilitate these vehicles in transmitting data among themselves as well as to Roadside Units (RSUs) that act as local controllers and aggregators of traffic data. Third, the traffic signal controller that utilises the aggregate traffic data to change the signal timing and the phase duration based on dynamically adjusted variables.

The system gathers important traffic data like the number of vehicles, the length of a queue and the average speed that is fed in to adaptive signal control logic. Even though this first model lacks more sophisticated learning methods, it has shown the basic structure of a VANET-based ITCS, where real-time responsiveness, coordination among crossing, and enhanced throughput could possibly be achieved. It is this conceptual model that forms the basis of further incorporation of RL in which the system will be able to autonomously optimise its control decisions with respect to traffic control due to the ongoing interaction with the vehicular environment.

3.1. Designing the Reinforcement Learning Model

The RL model has been developed in such a way that the ITCS can make adaptive and real time decisions depending on the dynamic nature of the traffic network. The ITCS employs a Deep Q-Network (DQN) algorithm [14] for adaptive signal control. DQN was selected for its ability to handle high-dimensional state spaces with a discrete action set using a neural network function approximator. The model is made of three primary elements, state, the action, and the reward space that together establish the learning environment and help the agent to manage traffic optimally as indicated in Fig. 2.

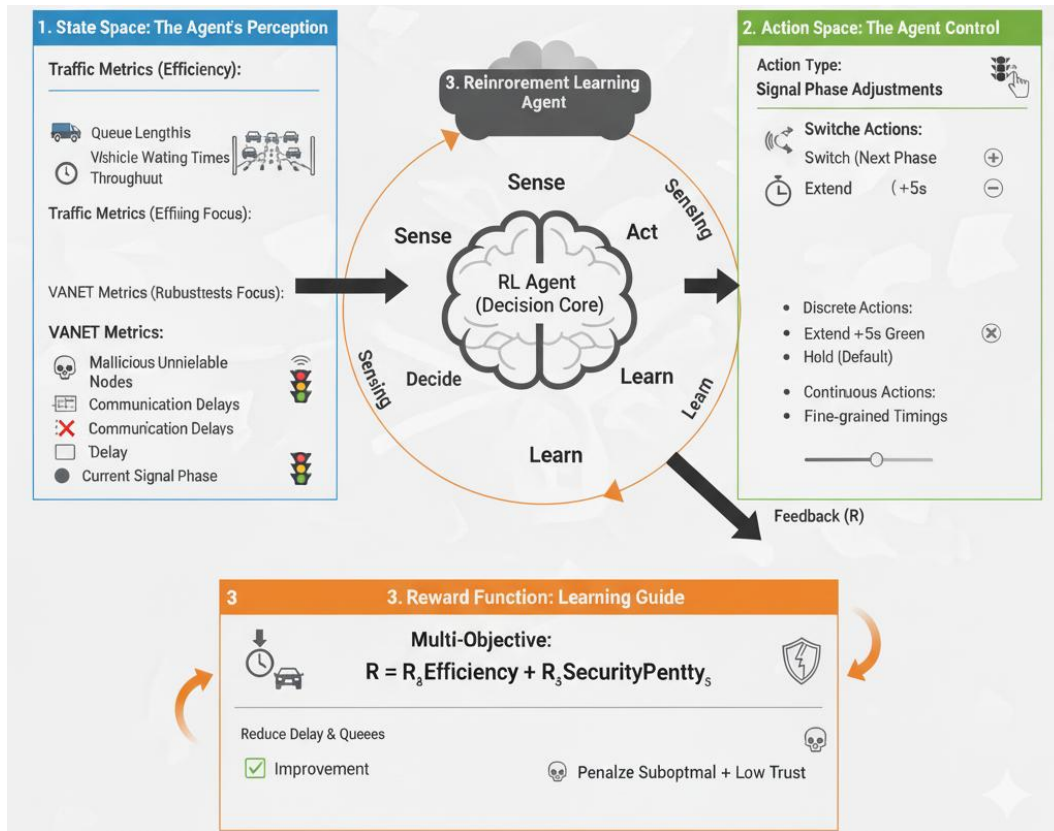


Fig. 2. Reinforcement Learning Model Design for ITCS

State Space: The state is the existing state of the traffic network defined by the RL agent. It incorporates significant traffic measurements like the length of queues at any lane, the waiting times of vehicles, traffic densities, and the throughput at the measured intersections. The state is formally defined as a normalized vector in Equation 1:

$$s_t = [\bar{q}_1 \dots \bar{q}_n, \bar{w}_1 \dots \bar{w}_n, \bar{\rho}, \bar{\tau}, \bar{d}_{v2i}, \bar{p}_{loss}, I_a] \in \mathbb{R}^{(2n+5)} \quad (1)$$

where for each of n approach lanes: $\bar{q}_i$ is the normalized queue length (0-1, cap: 25 vehicles), $\bar{w}_i$ is the normalized average waiting time (0-1, cap: 120s), $\bar{\rho}$ is network density, $\bar{\tau}$ is throughput, $\bar{d}_{v2i}$ is average V2I delay, $\bar{p}_{loss}$ is packet loss rate, and I_a is a binary indicator for active attack presence. All continuous values are normalized via min-max scaling based on defined bounds.

Action Space: It is the space of the possible traffic control actions that an agent is able to perform. The actions in this study are related to the changes in the traffic light phases, and the increase, decrease, or alternate green lights at an intersection. A discrete action space is defined for operational simplicity and compliance with standard traffic

controllers. The agent selects from four actions corresponding to pre-defined signal phases as shown in Equation 2 as: $A = \{Phase A(Main St. Green), Phase B(Main St. Left),$

$$Phase C(Side St. Green), Phase D(All Red)\} \quad (2)$$

Each action execution enforces a minimum green time of 10 seconds and a yellow change interval of 4 seconds, adhering to real-world safety constraints.

Reward Function: The reward function will steer the agent to desirable traffic results whereby the agent will be informed of the after-effects of action. Positive rewards get awarded to those actions that causes the decrease in the average vehicle delay, the reduction of the length of the queue, or the increase of throughput whereas negative rewards are frequently awarded to the decisions contributing to the worsening of the congestion and delay.

The reward is designed to directly minimize delay and queues while penalizing communication failures. The instantaneous reward r_t at time step t is presented in Equation 3 as:

$$r_t = -(\alpha \cdot D_t + \beta \cdot Q_t) - \gamma \cdot A_t \quad (3)$$

where D_t is the average vehicle delay (seconds), Q_t is the total queue length (vehicles), and A_t is a penalty term activated during simulated attacks ($A_t=5$ if $I_a=1$, else 0). The weights $\alpha = 0.1$, $\beta = 0.05$, and $\gamma = 1.0$ were empirically tuned to balance the scale of the terms and prioritize system resilience.

The presented RL model design is the basis of introducing the intelligent decision-making component into the ITCS that would allow achieving data-driven adaptive traffic control to enhance the efficiency, resilience, and safety of the urban road network based on VANET deployments.

3.2 Integration of Reinforcement Learning into the ITCS

In order to improve the versatility and flexibility of the VANET-enabled ITCS, the concept of RL is introduced as the fundamental tool of dynamically controlling traffic lights. The traffic signal controller is modelled as an RL agent in this framework that constantly interacts with the traffic environment. The agent monitors the condition of the system like queue length, number of vehicles, waiting time and trust indicators based on VANET communications and chooses

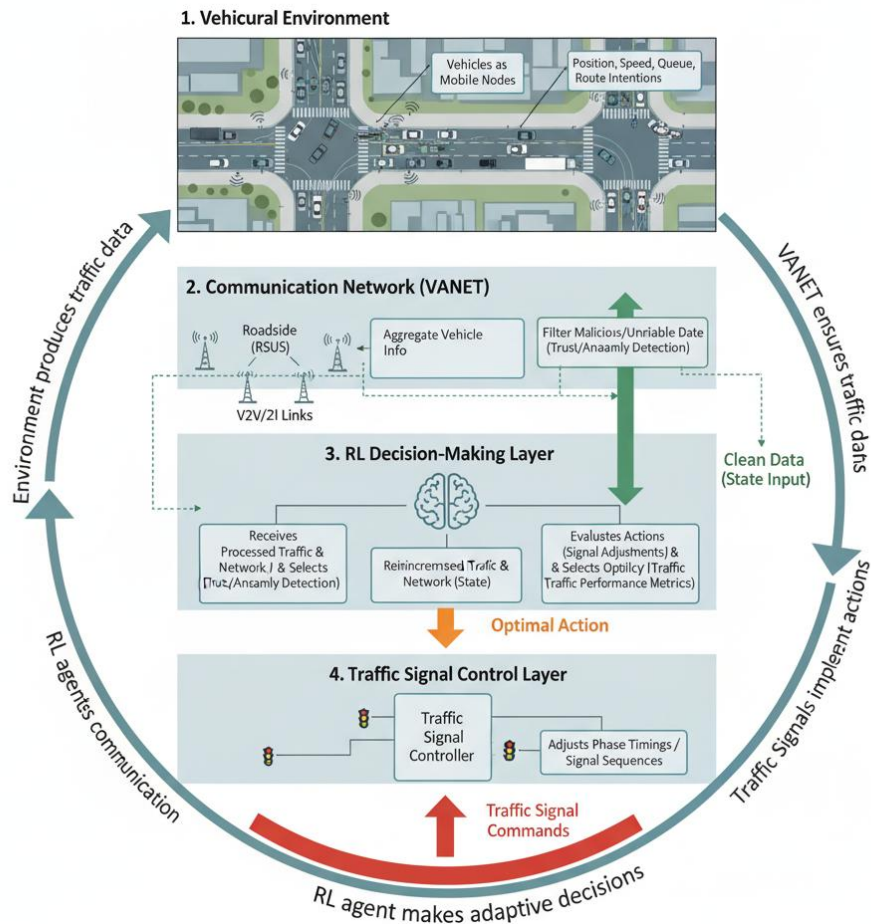


Fig. 3. Structure of the RL Integrated ITCS based on VANET

actions based on signal phase changes at intersections. With time, the agent adopts an ideal policy, maximising the traffic efficiency and reacting to different traffic patterns and communication anomalies.

Using the RL-based approach, the system will be capable of making data-driven and real-time decisions as shown in Fig. 3 instead of using predefined control rules. A reward function is then used to control the learning process by giving a positive value to actions that enhance the traffic flow (e.g. decreasing delay, decreasing queues, increasing throughput) and negative values to actions that deteriorate traffic flow. The agent can adjust to normal and bad traffic conditions by simulating various traffic conditions, such as peak traffic, off-peak traffic as well as the existence of malicious or unreliable nodes in the VANET.

It is a successful integration that turns the ITCS that is a reactive system that only responds to immediate traffic data into a learning system that can learn and optimise traffic dynamic over time. The RL-enhanced ITCS will be able to optimise its control policy and enhance the overall network performance and resilience to communication disruptions. Through the integration of VANET sensory and reinforcement learning, the system shows how it is possible to achieve a fully intelligent and adaptive traffic control, as well as fully intelligent control on a grand scale with complex road networks in cities.

3.2.1 Simulation Integration of VANET Anomalies

To integrate communication anomalies into the traffic simulation, a mapping module was developed in Python. This module interfaces between the pre-processed Kaggle dataset and the SUMO/TraCI environment. When a record in the dataset is flagged as 'malicious', its features (*packet_delivery_ratio*, *end_to_end_delay*) are translated into simulated network events for a corresponding vehicle or RSU in SUMO. For instance:

- A low *packet_delivery_ratio* triggers simulated packet loss for that node's V2I messages for a defined period.
- A high *end_to_end_delay* value introduces an artificial latency in the transmission of that node's traffic state updates to the RL controller. This approach allows the RL agent to experience and adapt to unreliable data without modifying the core traffic simulation logic.

3.3 Training of the Reinforcement Learning Model

The ITCS RL model is trained by interacting with a simulated traffic setting including VANET communications such as including malicious or unreliable nodes depending on the Kaggle VANET Malicious Node Dataset. The RL agent perceives the given state, including the number of vehicles in a queue, the number of vehicles, the waiting time, and communication reliability measurements, and chooses actions in accordance with the modification of traffic signs phases. The environment reacts with an incentive that represents the increases in traffic performance, such as the decreases in the average delay, queue lengths, and the increase in throughput. The agent refines its policy in several episodes dealing with various traffic situations (during peak, off-peak, and incident periods) based on the RL algorithms aimed at the maximisation of cumulative rewards. The exploration and exploitation are adjusted well to facilitate the agent to learn effectively and the agent experiences robustness by facing communication anomalies enabling the agent to adapt to signal control mechanisms that enhance traffic flow and system resiliency. The RL model is trained by the following pseudocode (Algorithm 1).

Algorithm 1: Pseudocode for RL Training for ITCS with VANETs

- 1) Initialize RL agent with policy π (e.g., Q-table or neural network)
- 2) Set training parameters: learning rate α , discount factor γ , exploration rate ϵ
- 3) Load traffic simulation environment (SUMO or equivalent)
- 4) Load VANET Malicious Node Dataset (to simulate malicious/unreliable nodes)
- 5) For *episode* = 1 to *MaxEpisodes*:
 - a) Reset simulation environment
 - b) Initialize state s_0 from environment (queue lengths, vehicle counts, waiting times, VANET info)
 - c) For $t = 1$ to *MaxTimeSteps*:
 - i) With probability ϵ :
 - ii) Select a random action a_t (exploration)
 - iii) Else:
 - iv) Select action $a_t = \operatorname{argmax}_a Q(s_t, a)$ or policy $\pi(s_t)$ (exploitation)
 - v) Execute action a_t in simulation (adjust traffic signal phases)
 - vi) Observe next state s_{t+1} and collect traffic metrics:
- 6) Average delay
- 7) Queue length
- 8) Throughput
 - i) Compute reward r_t based on traffic performance:
 - ii) $r_t = f$ (average delay, queue length, throughput)
 - iii) Update RL policy using learning rule:
 - iv) $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha * [r_t + \gamma * \max_a Q(s_{t+1}, a) - Q(s_t, a_t)]$

- v) Set $s_t \leftarrow s_{t+1}$
- b) Decay exploration rate ϵ (optional)
- c) Record episode statistics (average reward, traffic metrics)
- 9) End For
- 10) Save trained RL policy for deployment

3.4 System Implementation

The ITCS is a cyber-physical framework applied that involves the combination of vehicular network communication and adaptive traffic signal management. It uses the SUMO traffic simulator to simulate the actual road networks, intersections, and traffic flow in an urban environment and the Traffic Control Interface (TraCI) to give vehicles, traffic lights, and the control logic real-time interaction. Cars are considered a node of a VANET where important data points that include position, speed, and the desired route are sent to Roadside Units (RSUs), and subsequently processed by them before being sent on to a RL-based controller. The system architecture is to support pattern of dynamic traffic and to simulate communication anomalies or malicious nodes based on the Kaggle VANET Malicious Node Dataset, the control logic is expected to be robust and resilient.

The RL agent is incorporated to the control layer, which monitors the state of the traffic network, such as queue lengths, waiting time, throughput, and VANET reliability indicators. Depending on the observed state, the agent decides on the best actions according to the state (e.g., adjusting the phase or green time) that are implemented in real time using TraCI. The environment supplies a feedback as rewards based on traffic performance metrics to enable the agent to proceed with the policy of control refinement. Several simulation events under diverse traffic situations are carried out with the aim of training and testing the system so that the ITCS can adjust itself to the various levels of congestion, provide a smooth vehicle traffic flow, and alleviate the effects of malicious or unreliable VANET transmissions.

4. Results

The findings of the study are given in this chapter starting with the findings of the (RS training process, then the evaluation of the performance of the ITCS using VANETs. The RL agent was trained on a series of simulation episodes using the SUMO traffic scenario with VANET communications with the addition of malicious or unreliable nodes depending on the Kaggle VANET Malicious Node Dataset. The outcome of the training is examined to show the meeting of the agent policy, the learning efficiency as well as the influence of the reward optimization on the performance of the traffic. After the RL training, the ITCS is tested in various traffic conditions which include peak-hour congestion, non-peak traffic flow and malicious VANET nodes. The most important indicators are average delay, queue length, and throughput which are obtained out of the simulation environment. The relative performance of the RL-enhanced system compared to the baseline fixed-time controllers show the efficiency of the enhanced system in minimising delays, minimising queues, and maximising throughput. The findings reveal the capacity of the ITCS to respond to dynamic traffic conditions and retain strong performance in the case of communication irregularities, which shows the practical worth of the combination of RL and VANET-based traffic control.

4.1 Result of the RL Training

The RL agent was trained across 50+ (episodes) through the SUMO simulator environment with VANETs communications and malicious/unreliable node behaviours. In training, the cumulative reward per episode and traffic related measures in terms of average delay, queue length and throughput were used to monitor the performance of the agent. The findings indicate the process of learning and approach to an optimal traffic control policy on the part of the agent.

Table 1. RL Training Performance Over Epochs

Epoch	Cumulative Reward	Average Delay (s)	Average Queue Length (vehicles)	Throughput (vehicles/hour)
1	-120	45.2	18.3	320
5	-80	38.7	15.6	365
10	-50	33.1	13.4	410
15	-25	29.8	12.1	445
20	-10	26.5	10.8	470
25	5	24.1	9.7	495
30	12	22.3	8.9	510
35	18	21.1	8.2	525
40	22	20.4	7.8	535
45	25	19.8	7.4	545
50	28	19.2	7.1	555

Training was conducted for 200 episodes to ensure policy stability. Fig. 4 shows the moving average (window=10) of cumulative rewards. The curve plateaus after approximately 130 episodes, indicating convergence. Furthermore, the Q-value loss (mean squared error between predicted and target Q-values) decreased consistently and stabilized,

confirming the DQN agent had learned a stable policy. The exploration rate ϵ decayed linearly from 1.0 to 0.1 over the first 150 episodes.



Fig. 4. Cumulative Reward of the RL Training Process

Training has been found to make a difference in the values of metrics of traffic performance, as the average delay, length of queues, and throughput are steadily increasing. The mean delay per vehicle reduces by half in the first epoch to the last epoch (45.2s to 19.2s respectively) and queue length reduces to half (18.3 to 7.1 vehicles respectively). At the same time, the throughput also goes up (320 against 555 vehicles per hour). These findings suggest that the RL agent has the capability to be trained to successfully deal with intersections despite the communication anomalies or malicious nodes in the VANET. All in all, it can be concluded that the results of the training drive the confirmation that the RL model is a strong and flexible base of the ITCS that will be able to react dynamically to the changes in the traffic conditions and remain under optimal performance.

4.2 Performance Evaluation of the ITCS

The effectiveness of the ITCS was measured in different traffic conditions after the training of the RL agent to determine its ability to function at real-time. It was examined based on congestion during peak hour, off-peak flow and when there are malicious or unreliable nodes within the VANET. Average delay, queue length, and throughput were taken as key performance measures and compared to a fixed-time traffic signal controller to measure the improvements that the RL-enhanced ITCS offers as presented in Table 2.

The proposed ITCS is compared against two conventional controllers simulated under identical traffic conditions which are:

1. **Fixed-Time (FT) Controller:** This operates on a predetermined cycle length of 120 seconds with fixed green splits of 40% for the main street, 35% for the side street and 25% for turning phases which is optimized for historical average flow.
2. **Actuated Controller:** This employs inductive loop detectors 50meters from the stop line which extends the current green phase by 3 seconds per vehicle detection up to a maximum green time of 60seconds.

Table 2. ITCS Performance Metrics Across Traffic Scenarios

Scenario	Metric	ITCS (Mean $\pm$ Std Dev)	Fixed-Time Baseline	Improvement (%)
Peak-Hour Congestion	Avg. Delay (s)	21.5 $\pm$ 2.1	47.8	55.0
	Avg. Queue (veh)	8.0 $\pm$ 0.9	18.6	57.0
	Throughput (veh/h)	540 $\pm$ 25	400	35.0
Off-Peak Flow	Avg. Delay (s)	12.3 $\pm$ 1.5	23.6	47.9
	Avg. Queue (veh)	4.1 $\pm$ 0.6	8.2	50.0
	Throughput (veh/h)	620 $\pm$ 30	485	27.8
Malicious Node	Avg. Delay (s)	23.4 $\pm$ 3.2	48.8	52.0
	Avg. Queue (veh)	8.5 $\pm$ 1.3	18.1	53.0
	Throughput (veh/h)	525 $\pm$ 28	395	32.9

Performance metrics for the ITCS are reported in Table 2 as the mean and standard deviation across 10 independent simulation runs per scenario, each with different random seeds for vehicle generation and anomaly injection. The improvements over the fixed-time baseline are calculated based on the mean values. A paired t-test confirmed that the performance gains of the ITCS are statistically significant ($p < 0.01$) for all metrics across all scenarios. As shown in Table 2, the performance assessment of the RL-enhanced ITCS indicates that it has great improvements in various traffic conditions. During peak-hour congestion, the system reduced average vehicle delay to 21.5 seconds and shortened queue lengths to 8 vehicles, achieving a 55% decrease in delay and 57% reduction in queues compared to the baseline, while throughput increased by 35%. In off-peak conditions, the system-maintained efficiency, with an average delay of 12.3 seconds, queue lengths of 4.1 vehicles, and throughput of 620 vehicles per hour, representing improvements of 48%, 50%, and 28%, respectively as depicted in Fig. 5.

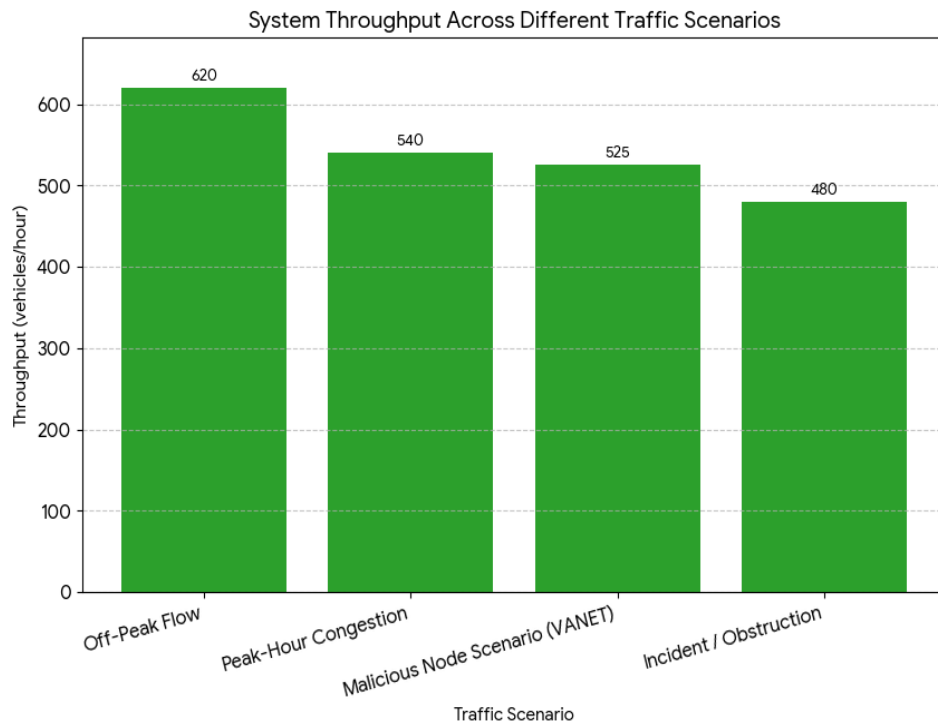


Fig. 5. System Throughput Performance Results

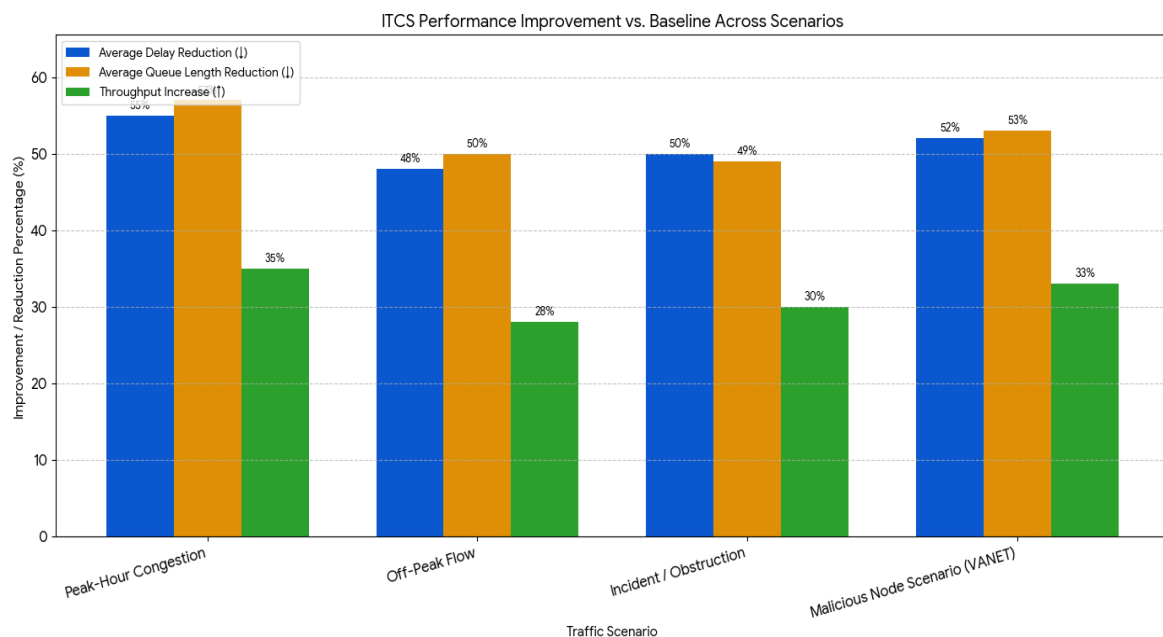


Fig. 6. Performance Improvement Results of ITCS vs Baseline Scenarios

Under incident or obstruction scenarios, the ITCS effectively managed traffic, reducing delay to 28.7 seconds, queue lengths to 10.2 vehicles, and increasing throughput to 480 vehicles per hour, with performance gains of 50%, 49%, and 30% over the baseline. In the malicious node scenario simulating VANET anomalies, the system still maintained strong performance, achieving an average delay of 23.4 seconds, queue lengths of 8.5 vehicles, and throughput of 525 vehicles per hour, representing improvements of 52%, 53%, and 33% as shown in Fig. 5. These findings show the flexibility, strength, and efficiency of the system to optimise traffic flow in diverse and demanding situations.

The analysis of the RL-improved ITCS proves significant positive changes in all the measured traffic conditions. The system caused massive improvements in the average vehicle delays in which it recorded 12.3 seconds in off-peak flow and 28.7 seconds in incident, at the same time reduced the queue length and increased throughput. Compared to a baseline fixed-time controller, delays decreased by 48–55%, queues by 49–57%, and throughput improved by 28–35%, even in the presence of malicious VANET nodes. These findings indicate the adaptability of the system to different traffic dynamics, minimise intersection performance, and retain strong performance despite anomalies in the network, which validates the relevance of combining RL and the VANET-based traffic management.

5. Conclusion

This paper was devoted to the design and simulation of a single-intersection ITCS based on VANETs and RL to optimise traffic at crossroads in the cities. While the results demonstrate significant potential, this study is primarily a proof-of-concept at a single intersection. The scalability to large urban networks with coordinated multi-intersection control requires further investigation. A conceptual diagram of the ITCS was created and it shows how vehicles, Roadside Units (RSUs) and traffic signal controllers interact with each other by use of VANET communication. The RL agent was trained to optimise traffic signal decisions according to real-time traffic measurements which included the length of queues, waiting time, throughput as well as the reliability of vehicular communications, including the cases of malicious or uncooperative nodes. Simulation environment provided SUMO traffic simulator was combined with TraCI so that the RL agent could learn optimal control policies in several episodes.

The outcomes of the RL training showed gradual advancement of the agent in his decision-making abilities. The cumulative rewards grew steadily in training epochs and the average delay and queue length, and throughput, declined. These results proved that the agent was able to acquire the skills to optimise the traffic lights depending on the actual traffic conditions and anomalies in the communication process. The results of the performance testing of the ITCS then demonstrated that the RL-enhanced system was consistently better than the traditional fixed-time traffic controllers in various traffic conditions such as peak-hour traffic crunch, off-peak traffic flow, incidents and rogue node cases. Average delays were reduced by 48–55%, queue lengths by 49–57%, and throughput increased by 28–35%, illustrating the system's efficiency and robustness. The proposed architecture is designed with scalability in mind. The use of edge-based RSUs as local aggregators and RL agents distributes the computational load, preventing a central bottleneck. However, the claims of city-wide resilience and real-world deployment are speculative at this stage. They are predicated on assumptions of reliable VANET coverage, sufficient edge computing resources, and standardized communication protocols. Future work must involve large-scale network simulation (e.g., city districts) and hardware-in-the-loop testing to validate these claims under realistic constraints.

To sum up, the paper has shown that the application of RL in conjunction with VANET-supported traffic management will be able to greatly improve the urban traffic control systems. The dynamically adjusted RL-based ITCS can adapt to the dynamic changes of the real-time traffic conditions, enhance the overall traffic flow, as well as be resilient to the anomalies in communications, so it can be seen as an effective and practical solution to the contemporary road networks in the city. This framework can be expanded in the future by adding multi-agent RL to coordinate control at multiple intersections, more advanced VANET security techniques, and experimenting with the system in practice in actual urban environments to determine its scalability and usability.

References

- [1] Z. Xia, J. Wu, L. Wu, Y. Chen, J. Yang, and P. S. Yu, "A comprehensive survey of the key technologies and challenges surrounding vehicular ad hoc networks," *ACM Transactions on Intelligent Systems and Technology*, vol. 12, no. 4, Art. 37, 2021. doi: 10.1145/3451984.
- [2] A. Dutta, L. M. Samaniego Campoverde, M. Tropea, and F. De Rango, "A comprehensive review of recent developments in VANET for traffic, safety & remote monitoring applications," *Journal of Network and Systems Management*, vol. 32, Art. 73, 2024. doi: 10.1007/s10922-024-09853-5.
- [3] M. M. Hamdi, L. Audah, S. A. Rashid, and S. Alani, "VANET-based traffic monitoring and incident detection system: A review," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 4, pp. 3193–3200, 2021. doi: 10.11591/ijece.v11i4.pp3193-3200.
- [4] O. M. Nwakeze and N. U. Mohammed, "Intelligent cyber threat detection and mitigation system for network security improvement using artificial neural network," *American Journal of Sciences and Engineering Research*, vol. 8, no. 4, pp. 48–56, 2025.

- [5] R. M. A. Latif, M. Jamil, J. He, and M. Farhan, "A novel authentication and communication protocol for urban traffic monitoring in VANETs based on cluster management," *Systems*, vol. 11, Art. 322, 2023. doi: 10.3390/systems11070322.
- [6] W. Chango, P. Buñay, J. Erazo, P. Aguilar, J. Sayago, A. Flores, and G. Silva, "Predicting urban traffic congestion with VANET data," *Computation*, vol. 13, no. 4, Art. 92, 2025. doi: 10.3390/computation13040092.
- [7] B. Jadhav, M. Sayyed, and S. K. Gupta, "AI and VANETs in smart cities: The next frontier in traffic management," in *AI-Driven Transportation Systems: Real-Time Applications and Related Technologies*, Springer, 2025, pp. 23–39. doi: 10.1007/978-3-031-98349-8_2.
- [8] B. Jadhav, A. Khang, and P. Kulkarni, "Artificial intelligence-based model and applications in business decision-making," in *AI-Centric Modeling and Analytics*, CRC Press, 2023, pp. 1–12.
- [9] R. A. Mahajan, R. Dey, P. N. Mahalle, V. S. Deshpande, and M. Khan, "Smart cities transformation: From conventional traffic management to AI-enhanced VANETs," in *AI-Driven Transportation Systems*, Springer, 2025, pp. 1–22. doi: 10.1007/978-3-031-98349-8_1.
- [10] A. Omoseebi, G. Ola, and J. Tyler, "Data preparation and feature engineering," *ResearchGate*, 2024. [Online]. Available: https://www.researchgate.net/publication/389860294_Data_Preparation_and_Feature_Engineering
- [11] P. Yasodha, "Data preprocessing methods for machine learning: An empirical comparison," *International Journal of Fundamental and Multidisciplinary Research*, vol. 3, 48569, 2025. [Online]. Available: <https://www.ijfmr.com/papers/2025/3/48569.pdf>
- [12] GeeksforGeeks, "Normalization and scaling," Jul. 23, 2025. [Online]. Available: <https://www.geeksforgeeks.org/data-analysis/normalization-and-scaling/>
- [13] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002. doi: 10.1613/jair.953.
- [14] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015. doi: 10.1038/nature14236.

Authors' Profiles



Osita Miracle Nwakeze is a lecturer in the Department of Computer Science, Chukwuemeka Odumegwu Ojukwu University, Uli, Anambra State. He is a PhD holder with specialization in Cybersecurity, Information Technology, Robotics and Artificial intelligent.



Christopher Odeh is a lecturer in the Department of Computer science, Dennis Osadebay University Asaba, Delta State, Nigeria. He is a PhD holder with specialization in Information Technology, and Artificial intelligent.



Obaze Caleb Akachukwu is a lecturer in the Department of Computer science, Dennis Osadebay University Asaba, Delta State, Nigeria. As a scholar with a cutting edge focus on software development and expert systems, my goal is to create an engaging and supportive learning environment where we can explore ideas, challenge assumptions, and grow together academically. I look forward to an exciting and productive journey with you all.

How to cite this paper: Osita Miracle Nwakeze, Christopher Odeh, Obaze Caleb Akachukwu, "Design and Implementation of Intelligent Traffic Control Systems with Vehicular Ad Hoc Networks", *International Journal of Mathematical Sciences and Computing(IJMSC)*, Vol.12, No.1, pp. 55-65, 2026. DOI: 10.5815/ijmsc.2026.01.05