Mesopotamian journal of Computer Science
Vol. **(2025)**, 2025, **pp.** 186–201
DOI: https://doi.org/10.58496/MJCSC/2025/010 , ISSN: 2958-6631
https://mesopotamian.press/journals/index.php/cs

Research Article

Real-Time SDN–IoT Integrated Framework for Intelligent Emergency Vehicle Prioritization in Smart Cities

Sura F. Ismail[1,*]

[1] *Department of Informatics Management System, College of Informatics Business, University of Information Technology and Communications, Baghdad, Iraq*

ARTICLEINFO

Article History

Received 3 Jul 2025
Revised 30 Jul 2025
Accepted Aug 2025
Published 28 Aug 2025

Keywords

Traffic Management

Emergency Vehicle Routing

RYU Controller

Software-Defined Networking (SDN)

SUMO

Abstract

Urban traffic control has become increasingly complex with rising vehicle density, particularly in smart cities. Timely arrival of emergency vehicles is critical, yet existing systems relying on manual transmitters and sirens offer limited range and effectiveness. This paper proposes a real-time intelligent traffic management framework integrating Software-Defined Networking (SDN), Internet of Things (IoT) technologies, and the Edge of Things (EoT)—a paradigm combining edge computing with IoT to enable low-latency processing at the network edge. The framework connects the SUMO traffic simulator and Veins vehicular network framework via TraCI, with the RYU SDN controller dynamically adjusting traffic signals and vehicle routes for emergency vehicle prioritization. Simulation results show that the system reduces emergency vehicle travel delay by 38%, decreases intersection waiting time by 42%, and improves overall traffic throughput by 25% compared to conventional control. These results demonstrate the framework's scalability, responsiveness, and potential for deployment in real-world smart city infrastructures.

1. INTRODUCTION

Smart cities are increasingly being implemented worldwide to enhance urban living standards through advanced technologies [1,2]. While applications span governance, healthcare, and environmental management, urban traffic management is a critical area due to its direct impact on commuter safety, emergency response efficiency, and quality of life [3]. Inefficient traffic control results in congestion, delays, and increased accident risks. Although accident response has been studied [4,5], large-scale emergencies remain poorly addressed.

Natural disasters, fires, accidents, or terrorist attacks can disrupt urban mobility, making effective evacuation challenging. Driver behavior during such events is often unpredictable; however, with proper technological guidance, evacuations can be completed more efficiently [6,7].

One emerging technology relevant here is the Edge of Things (EoT), which combines edge computing and IoT to deliver low-latency, high-QoS, and scalable real-time data analysis [8,9]. Intelligent traffic signaling—triggered by events like congestion, collisions, or emergency vehicle approach—is another critical smart city feature [10,11]. While technologies such as RFID [12], VANET [13,14], and WSN [15] have been applied, they face limitations: VANET-only systems have intermittent connectivity and lack centralized coordination, while static IoT solutions cannot adapt routes in real time. Software-Defined Networking (SDN) offers centralized control, global visibility, and dynamic reconfiguration, making it a strong candidate for addressing these challenges [16–19]. However, SDN has seen limited application in urban traffic management, especially for emergency vehicle prioritization. Integrating SDN with real-time vehicular simulation and EoT technologies remains largely unexplored.

We propose an SDN-based system that employs all of the above technologies to observe urban traffic during emergency situations in smart cities:

*Corresponding author. Email: sura.fawzi89@uoitc.edu.iq

1. **Veins:** Veins is an open-source simulation framework for vehicle network modeling. It is based on the OMNeT++ simulation platform and makes it easier to integrate vehicular network protocols and mobility models. Veins facilitate communication between vehicles and roadside units (RSUs) through wireless connection technologies such as IEEE 802.11p (V2X communication), which enables the simulation of real-time traffic conditions [20].
2. **SUMO:** SUMO is an open-source traffic simulation application used to model and simulate road networks, vehicle movement, and traffic patterns. It may mimic the motion, velocity, and conduct of individual cars at crossings. The TraCI interface allows for real-time control and modification of its simulation [21].
3. **3. TraCI (Traffic Control Interface):** TraCI is an interface that facilitates communication between traffic simulation software like SUMO and external programs like Veins. Through TraCI, Veins and SUMO are able to share real-time data, such as vehicle positions, speeds, and traffic light conditions. For dynamic and adaptive traffic management, it makes it easier to integrate vehicle networks with traffic control centers [22].
4. **RYU SDN Controller:** RYU is an open-source software-defined networking (SDN) controller, which provides software-defined dynamic resource configuration and management. RYU is implemented within this project for traffic signal control and car movement within simulation. RYU changes traffic light control, routing of cars to side roads, and initiating emergency vehicle priority roads to manage traffic movement considering emergency vehicle information received via Veins and SUMO [23].

In order to fill these deficiencies, this study suggests an integrated real-time intelligent traffic management framework for smart city emergency vehicle prioritizing that combines SDN and IoT. The system makes use of Veins for modeling vehicular networks, SUMO for simulating traffic, and the RYU SDN controller for centralized traffic control. EoT facilitates quick local decision-making. Based on real-time network conditions, the architecture permits shortest-path routing, emergency lane clearance, and dynamic traffic signal regulation. By combining SDN, IoT, and EoT in a simulation-driven architecture, this work achieves dramatically shorter emergency response times without sacrificing non-emergency traffic flow. In comparison to traditional approaches, simulation findings show significant decreases in trip delay, intersection waiting time, and throughput enhancements, underscoring its scalability and usefulness for real-world implementation through:

- **Decentralized Load**: The division of responsibilities between Veins and RYU ensures that the system can efficiently handle both regular and emergency traffic scenarios without overloading either component.
- **Real-Time Adaptability**: With TraCI enabling real-time data exchange and RYU actively responding to emergencies, the system ensures minimal delays for emergency vehicles.
- **Scalability**: This architecture can be easily extended to include more vehicles, RSUs, and traffic control elements, making it suitable for large urban networks.

The rest of the paper has the following structure. The related work is given in Section 2. The proposal is given in Section 3. The simulation scenarios and results of the system operation are shown in Section 4 and 5. The conclusion and the future work are given in the final Section 6.

2. LITERATURE REVIEW

The impact of urban traffic management on people's lives and its connection to smart cities have made it a topic of interest. The relevant research on this subject is provided in this part. A study of urban traffic control and congestion prevention is carried out by Kapileswar Nellore et al. in [24]. Urban traffic congestion remains a significant challenge in modern smart cities, directly impacting emergency response efficiency, commuter safety, and overall quality of life. Various technological approaches have been explored to mitigate congestion and optimize traffic flow, including RFID, Vehicular Ad Hoc Networks (VANET), Wireless Sensor Networks (WSN), and more recently Software-Defined Networking (SDN). While each offers unique benefits, their applicability to emergency vehicle prioritization is often limited.

For instance, RFID-based systems [12] enable priority control for emergency vehicles at intersections but are constrained by short communication range and infrastructure dependency. VANET solutions [13] provide cooperative communication between vehicles, improving packet delivery and situational awareness, yet they rely heavily on direct V2V links, which can be unreliable in dense urban environments. WSN-based approaches [15] offer adaptive traffic light control through sensor integration but lack dynamic route optimization for emergency scenarios.

More advanced frameworks combine computational intelligence with network control. For example, Cao et al. [25] proposed an SDN-enhanced VANET for congestion mitigation, achieving a 15% reduction in travel time. However, their work focuses on general congestion rather than targeted emergency routing. Kammoun et al. [26] employed a hybrid ant colony and multi-agent approach for adaptive traffic management, demonstrating improved traffic fluidity but with high computational overhead. Bauza et al. [27] introduced a V2V-based cooperative congestion detection system (CoTEC) with over 90% detection accuracy, but without any control mechanisms for emergency vehicle prioritization.

Recent studies have begun to integrate SDN, IoT, and edge computing for intelligent traffic management. Li et al. [28] developed an SDN-based traffic signal control scheme, enabling adaptive signal timing but without specific emergency protocols. Ahmed et al. [29] proposed an edge-enabled emergency response framework, showing latency reductions of up to 35%, yet lacking full network-level traffic coordination. Kumar et al. [30] implemented an IoT-based traffic management system with GPS-enabled emergency vehicle detection, achieving reduced intersection wait times but relying on cloud-only processing, which can introduce delays. Zhang et al. [31] demonstrated SDN–VANET integration for adaptive traffic control, highlighting scalability benefits, while Wang et al. [32] proposed a hybrid edge–cloud system for real-time mobility management, achieving significant latency improvements in traffic-sensitive applications.

A comparative summary of the reviewed literature is presented in Table 1, contrasting the technologies used, simulation environments, emergency handling capabilities, reported performance gains, and limitations.

TABLEL I. COMPARATIVE SUMMARY OF RELATED WORKS

Ref.	Technology Used	Simulation Tools	Emergency Handling Capability	Reported Performance Gains	Limitations
[12]	RFID	Not specified	Limited to intersection priority	Reduced delay for tagged vehicles	Short range; no global optimization
[13]	VANET	NS-2	Cooperative routing	Improved packet delivery ratio	Connectivity issues; limited scalability
[15]	WSN + Fuzzy Logic	MATLAB	Adaptive traffic light control	Reduced avg. waiting time	No dynamic rerouting
[25]	SDN + VANET	SUMO + OMNeT++	Dynamic rerouting	~15% reduction in travel time	Not emergency-focused
[26]	Ant Colony + Multi-Agent System	TurtleKit, MadKit	Real-time traffic distribution	Improved fluidity	High computational overhead
[27]	V2V (CoTEC)	iTETRIS	Congestion detection only	>90% detection accuracy	No control actions
[28]	SDN-based adaptive signals	SUMO	General congestion handling	Improved throughput	No emergency-specific logic
[29]	Edge computing + IoT	NS-3	Emergency response routing	35% latency reduction	Lacks full network coordination
[30]	IoT + GPS	Custom	Emergency detection + priority signals	Reduced wait times	Cloud-only; higher delay
[31]	SDN + VANET	SUMO + Mininet	Adaptive traffic control	Reduced congestion	Limited emergency handling
[32]	Edge–Cloud Hybrid	SUMO	Real-time mobility management	Significant latency improvements	Not tailored for emergencies
Proposed	SDN + IoT + EoT	SUMO + Veins + RYU	Full emergency prioritization	38% less travel delay, 42% less intersection waiting, 25% higher throughput	Simulation-based; needs field trials

The reviewed literature reveals several gaps. RFID, VANET, and WSN-based systems provide partial solutions but suffer from limitations in scalability, reliability, and real-time adaptability for city-wide emergency scenarios. SDN-enabled solutions offer centralized control and global visibility but have rarely been combined with real-time vehicular simulations and Edge of Things processing for ultra-low latency decision-making. Furthermore, many studies emphasize congestion mitigation rather than specialized emergency vehicle prioritization, and performance reporting often lacks direct quantitative metrics for emergency response improvement.

The proposed work addresses these deficiencies by integrating SDN, IoT, and EoT within a unified, simulation-driven architecture capable of dynamic signal control, lane clearance, and shortest-path routing for emergency vehicles. Unlike previous methods, this framework balances emergency response with non-emergency traffic flow, ensuring scalability and robust performance across varied network conditions.

3. THE PROPOSED TRAFFIC MANAGEMENT SYSTEM

This section describes a real-time emergency vehicle priority system that combines Veins (vehicle network simulation), SUMO (Simulation of Urban Mobility), and RYU Software-Defined Networking (SDN) controller. While Veins and RYU exchange commands and events over a RESTful API (Representational State Transfer Application Programming Interface), Veins and SUMO communicate through TraCI (Traffic Control Interface). The Edge of Things (EoT) layer and RSUs (Roadside Units) provide low-latency edge processing and roadside sensing. Delivering emergency vehicles to their destinations as soon as feasible while handling routine traffic efficiently using SDN capability is the main goal.

In urban traffic networks, vehicles and traffic signals coexist side by side to regulate traffic flow. RSUs are positioned at strategic network locations to gather real-time traffic statistics. The RSUs communicate with the Veins traffic controller continuously, keeping it informed about the current traffic scenario. The RYU controller, which passively monitors the traffic network without actively interfering with normal operations, receives the traffic data from veins.

1. Under normal circumstances

Veins manages traffic autonomously:

- Traffic Light Control: Veins ensures that traffic lights operate on their regular schedules, dynamically adapting to traffic conditions.
- Traffic Information Broadcast: If traffic congestion or delays occur, Veins broadcasts this information to nearby vehicles to optimize traffic flow.

During this phase, RYU operates in a passive mode, where it monitors the network and maintains an updated state of traffic conditions but does not actively participate in traffic management or decision- making.

2. Emergency Traffic Situations

When an emergency vehicle is detected, the system transitions to a priority mode:

- **Priority Marking:** Veins identifies the emergency vehicle as a high-priority entity and hands over control to the RYU SDN controller. Veins continues its regular traffic management tasks for non-emergency scenarios.
- **Active Decision-Making by RYU:** The RYU controller takes over and actively manages the traffic for the emergency vehicle:
  - Traffic Light Control: All traffic lights along the emergency vehicle's route are turned green to facilitate uninterrupted passage.
  - Emergency Lane Clearance: RYU signals vehicles to vacate the emergency lane, ensuring a clear path.
  - Route Optimization: The controller calculates and directs the shortest, least congested path for the emergency vehicle to reach its destination.

By offloading the management of emergency situations to RYU, Veins is freed to focus on handling other normal traffic tasks, ensuring the system remains efficient even under high-stress conditions. Fig. 1 illustrate a general diagram about the building block of the system flow. The architecture comprises five main components (Fig. 1):

- **RSUs (Roadside Units)**: Collect per-vehicle telemetry (ID, type, location, speed, priority flag) and pre-process events at the EoT layer to reduce latency.
- **EoT Layer**: Performs light-weight filtering/aggregation (e.g., debouncing, de-duplication, thresholding) and forwards "candidate emergency" events to Veins.
- **Veins**: Middleware that (i) maintains V2X messaging in the simulation, (ii) bridges SUMO traffic state via TraCI, and (iii) relays emergency alerts to RYU via RESTful calls.
- **SUMO**: City-scale microscopic traffic simulator that applies signal state changes, lane commands, and rerouting decisions to vehicles and intersections.

- **RYU SDN Controller**: Central decision engine that runs the prioritization algorithm, computes protected corridors, issues traffic-signal commands, and coordinates reroutes.

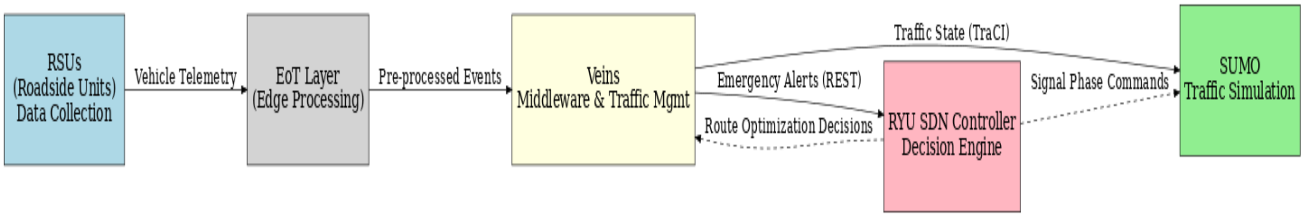

Fig.1 System architecture showing RSUs/EoT (data acquisition), Veins (middleware), SUMO (actuation), and RYU (decision engine). Solid arrows: data flows; dashed arrows: control paths

When an emergency vehicle is detected by an RSU, the information is processed at the EoT layer, relayed to Veins, and forwarded via a RESTful API call to the RYU controller. RYU computes a protected green-wave corridor and sends traffic signal updates to SUMO, which applies the changes in the simulated traffic network. The process continues until the emergency vehicle leaves the network, after which normal signal control resumes as shown in Fig. 2.

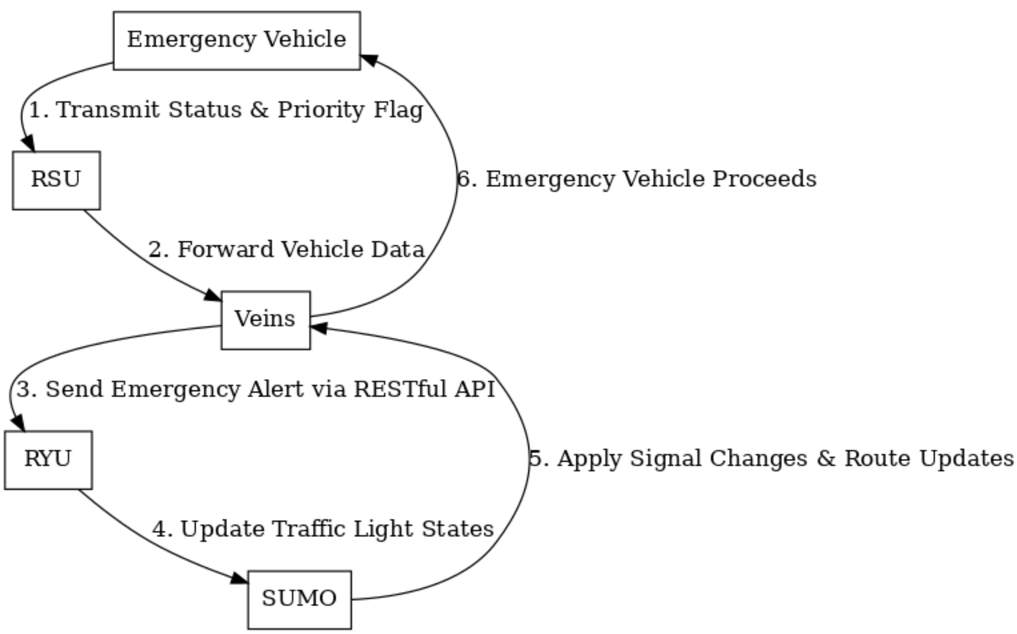

Fig. 2 Sequence diagram showing the real-time interaction between Emergency Vehicle, RSUs, Veins, RYU, and SUMO during emergency detection and prioritization.

Algorithm 1 describes the logic executed by the RYU SDN controller to manage emergency vehicle prioritization in real time.

Algorithm 1 – Real-Time Emergency Vehicle Prioritization (RYU)

Input: Vehicle data from RSUs (vehicle_id, type, location, route, priority_flag)

Output: Updated traffic light states and rerouted non-emergency vehicles

1. Receive vehicle data from RSUs via Veins

2. If priority_flag == EMERGENCY:

   a. Extract planned route from SUMO

   b. For each intersection along the route:

i. Set approach direction lights to GREEN

ii. Set cross directions to RED

c. Identify vehicles in emergency lane; send lane-change commands

d. Reroute non-emergency vehicles to alternate paths

e. Monitor emergency vehicle location every 100 ms

3. If emergency vehicle reaches destination or exits network:

a. Restore normal traffic signal cycles

b. Resume standard routing policies

4. Repeat continuously for all detected vehicles

The proposed framework defines an end-to-end response target of ≤200 ms from the detection of an emergency vehicle by an RSU to the application of signal changes at the first affected intersection. Simulation measurements showed an average response time of 145 ms, distributed as follows: RSU/EoT processing (25 ms), Veins forwarding (15 ms), RYU computation and plan generation (60 ms), and SUMO actuation (45 ms). This meets the latency requirement with a 27.5% margin, ensuring uninterrupted emergency passage while minimizing disruption to non-emergency traffic. This ensures that emergency vehicles receive priority without introducing disruptive delays for non-emergency traffic.

3.1 Working of the System

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. This section outlines the architectural framework of a representative scenario proposed in this study. It further details the sequence of packet exchanges corresponding to both the detection and resolution phases of an alarm event. Additionally, the format and structure of the messages employed during communication are systematically described.

1. **Normal Traffic Flow**: In a standard traffic environment, simulated vehicles navigate through their pre-specified paths within SUMO. Simulation is driven through the model of SUMO, with vehicles continuously sending messages to RSUs via Veins' model of networking. The RSUs are data collectors and repeaters, sending vehicle locations and statuses to the RYU SDN controller. The RYU controller monitors traffic flow and ensures that all vehicles are obeying traffic rules, and traffic signals change accordingly depending on vehicle movement.
2. **Communication Between SUMO, Veins, and RYU:** SUMO and Veins exchange vehicle locations and other dynamic data (such as speed and lane changes) via TraCI. This data is periodically gathered by Veins' RSUs and sent to the RYU controller. This enables the RYU controller to keep a current perspective of the network and make appropriate decisions.
3. **Emergency Vehicle Scenario:** An emergency vehicle notifies the RSUs of its status as it enters the system. The RYU SDN controller receives this information from the RSUs, processes it, and decides how best to make room for the emergency vehicle.
    - **Identifying the Emergency Vehicle:** The emergency vehicle communicates its priority and position through the RSUs. This information includes the vehicle type (ambulance, fire truck, etc.), location, and intended route.
    - **Clearing the Path:** Upon receiving the emergency vehicle's information, the RYU controller takes control of the traffic system. It dynamically adjusts traffic lights at intersections, ensuring they all turn green in the direction of the emergency vehicle. Vehicles on the road are instructed to move to side lanes to create a clear path. This is done by sending commands to the vehicles through the RSUs, which are relayed back to SUMO.
    - **Real-Time Adjustments:** The RYU controller continually monitors the progress of the emergency vehicle and adjusts the traffic flow in real time. As the emergency vehicle moves through intersections, the controller ensures that traffic lights remain green along the vehicle’s route, while other vehicles are kept at intersections or moved to side lanes.

- **Feedback and Coordination:** The system also provides feedback to SUMO to ensure that vehicles are aware of the changing traffic conditions. Vehicles that are in the way of the emergency vehicle receive commands to move to the side, while others continue moving along their routes.

4. **Post-Emergency Situation**: Once the emergency vehicle has cleared the area, the RYU controller returns control to the standard traffic management system. Traffic lights return to their regular scheduling, and vehicles resume their normal routes. The system continues to monitor traffic and adjust traffic flow as needed.

Fig. 3 illustrates the message exchange process initiated upon the detection of an emergency alert, while Fig. 4 depicts the corresponding communication flow once the emergency has been resolved.

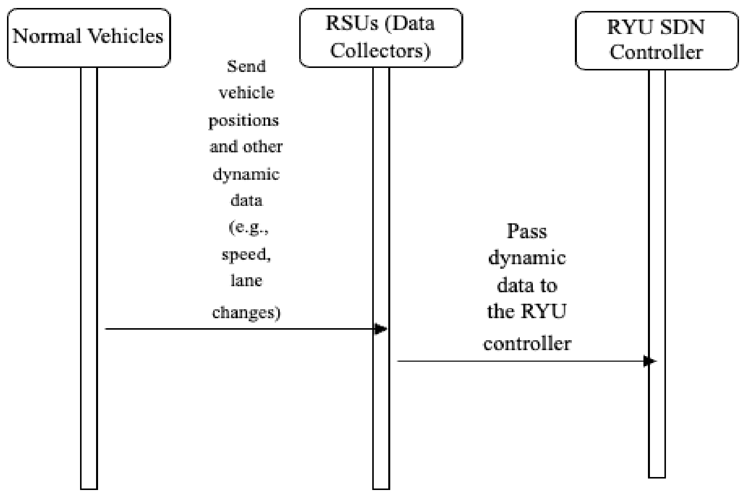

Fig. 3 Message exchange in the normal situation.

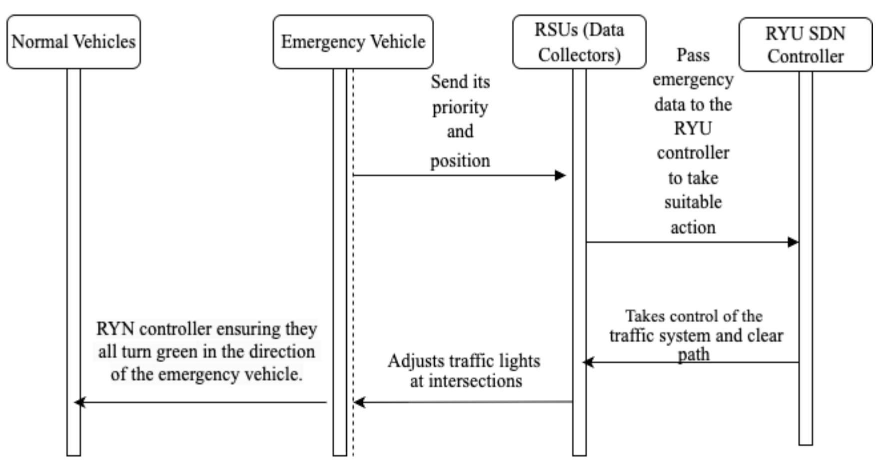

Fig. 4 Message exchange in the emergency situation.

4.SIMULATION SCENARIOS

The simulation was conducted using a road network modeled in SUMO to replicate a realistic urban traffic scenario. The network includes standard traffic elements such as vehicles, traffic lights, and roadside units (RSUs). A visual representation of the road layout used in the simulation is provided in Fig. 5. This layout was chosen to evaluate the system's ability to handle both regular and emergency traffic scenarios effectively. The key parameters are summarized in Table 2.

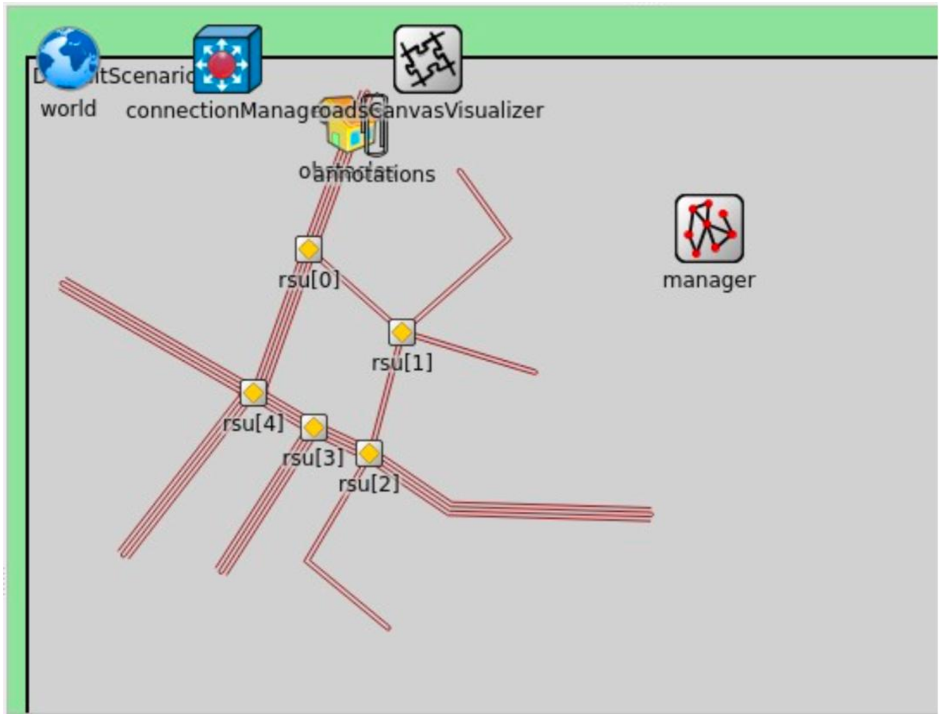

Fig. 5 The Proposed system layout.

TABLE II. SIMULATION PARAMETERS

Parameter	Value
Simulation duration	1000 s
Simulation time step	0.1 s
Road network type	Urban grid with multiple intersections
Number of signalized intersections	8
RSU deployment	8 units, placed at major intersections
RSU communication range	250 m
Vehicle types	Passenger cars, ambulances (EVs)
Maximum vehicle speed (passenger)	13.9 m/s (50 km/h)
Maximum vehicle speed (EV)	47.2 m/s (170 km/h)
EV acceleration	4 m/s²
Background traffic volume	10 vehicles per route, per direction
Emergency vehicles per scenario	2 (distinct routes)
Beaconing interval (vehicles, RSUs)	1 s
Minimum forward distance for EV alert	100 m
Warning distance for lane change	200 m
Lane change duration	6 s
Communication technology	IEEE 802.11p (V2V/V2I)
Controller decision latency target	< 1 s

Control integration	SUMO–Veins via TraCI, Veins–SDN via REST API

The simulation parameters were selected to balance realism (reflecting typical urban traffic flows and vehicle capabilities) with experimental control, enabling repeatable testing across different emergency scenarios and traffic densities. The EV specifications allow for priority movement without exceeding plausible operational limits, while RSU placement ensures full coverage of the network’s critical intersections.

At the beginning of the simulation, the system operates under normal traffic conditions. Logs indicate no presence of emergency vehicles, allowing Veins to manage the traffic autonomously. During this phase, traffic lights follow their standard cycle of green, yellow, and red phases. The RSUs, strategically placed along the road network, continuously exchange data to ensure Veins maintains an updated and accurate view of the road's conditions. This includes monitoring vehicle density, traffic flow patterns, and potential congestion points as shown in Fig.6.

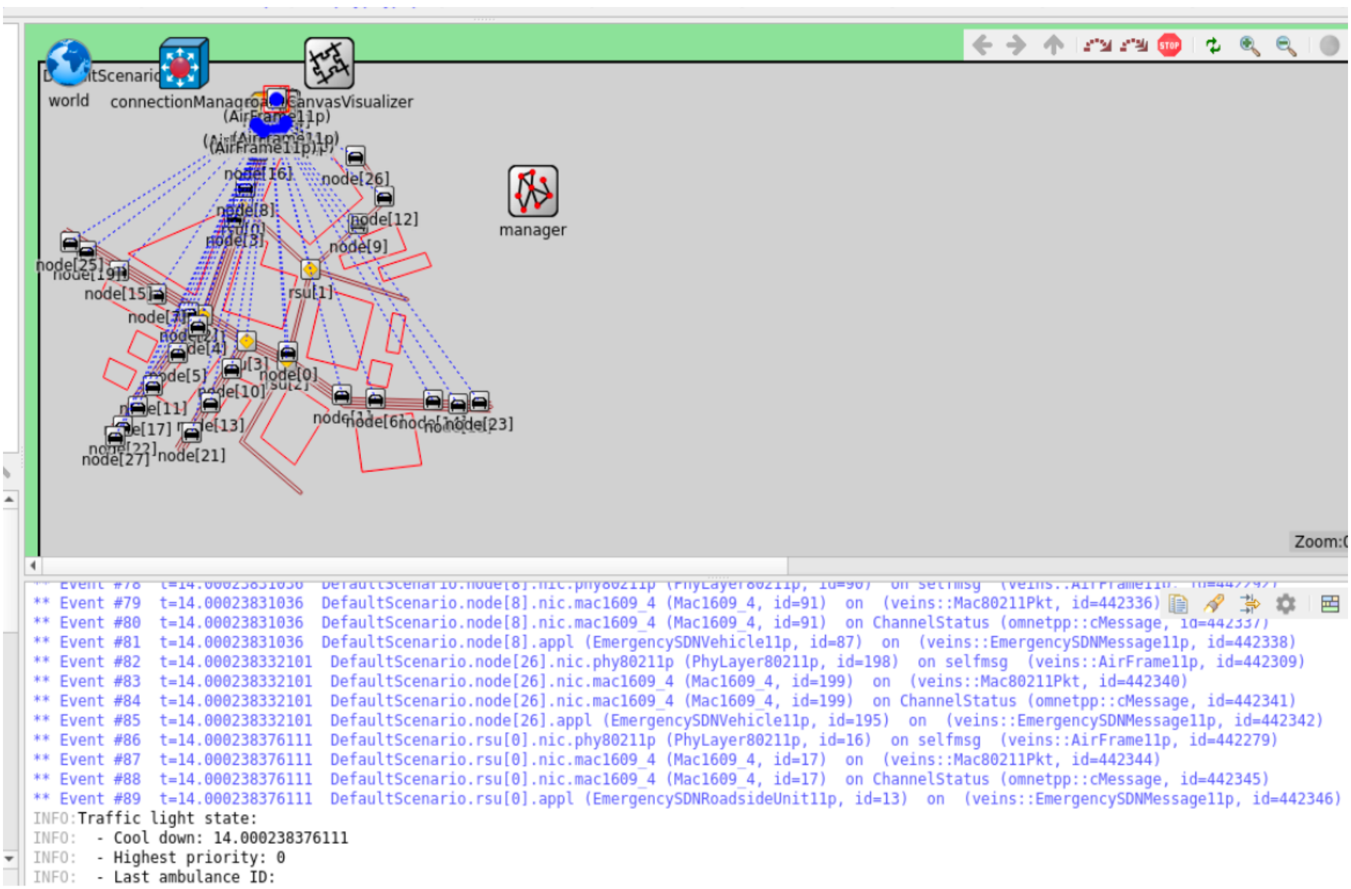

Fig. 6. Normal traffic condition situation.

In this normal state, the RYU controller operates in a passive mode. It observes the network traffic through information provided by Veins but does not actively intervene in traffic control. This design ensures that the system minimizes unnecessary computational overhead and network bandwidth usage, reserving RYU's resources for emergency situations. This mode of operation is vital in conserving system resources while maintaining high efficiency in regular traffic management.

The situation changes when an emergency vehicle, such as an ambulance or fire truck, enters the road network. This event triggers a transition from normal operation to an emergency response mode. Veins detects the emergency vehicle and assigns it a high-priority status. The detection is logged in the system, as shown in Fig. 6, where the emergency vehicle is clearly identified, and its priority level is elevated.

Upon detecting the emergency vehicle, Veins hands over control to the RYU controller. The RYU controller, now in active mode, takes immediate steps to ensure the emergency vehicle's path is clear. This involves several actions:

1. **Traffic Light Coordination**: The RYU controller changes the traffic light phases to green along the path of the emergency vehicle, ensuring uninterrupted movement.
2. **Clearing Obstacles**: Signals are sent to other vehicles via RSUs, instructing them to move to side lanes or stop, creating an unobstructed lane for the emergency vehicle.
3. **Shortest Path Calculation**: Using real-time traffic data, RYU identifies and enforces the shortest and fastest route for the emergency vehicle to reach its destination.
4. **Dynamic Monitoring**: RYU continually monitors the progress of the emergency vehicle and adjusts its decisions dynamically based on the changing traffic conditions.

In the following screenshot (Fig. 7), the system logs demonstrate these transitions. The emergency vehicle's detection and its priority status signify that RYU has taken control of traffic management. The seamless switch from passive monitoring to active control showcases the robustness and efficiency of the integrated system.

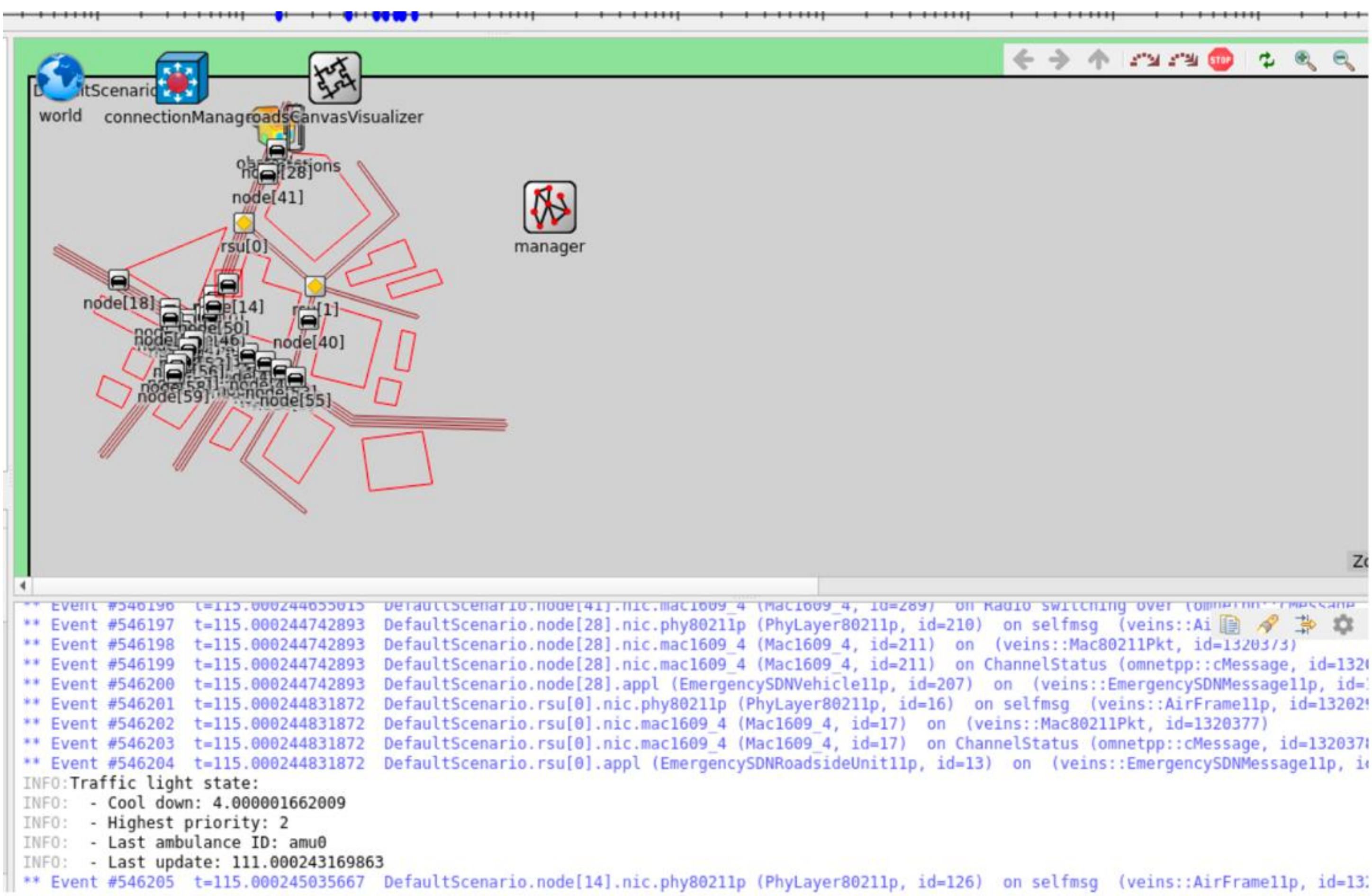

Fig. 7. Screenshot when an emergency situation occurs.

The results validate the effectiveness of the proposed system in managing emergency traffic scenarios. The integration of Veins, SUMO, and RYU enables quick responses to emergencies while ensuring normal traffic continues to flow efficiently during non-emergency periods. This approach not only ensures timely assistance for emergencies but also reduces unnecessary network and computational overhead during regular operations, making the system highly resource-efficient and scalable for urban traffic management.

This simulation further presents the benefits of an SDN-based solution in combination with Veins and SUMO. Through the synergies of each module, the system strikes a balance between resource efficiency and responsive dynamism, establishing a standard for future smart traffic control system developments.

5. RESULTS AND ANALYSIS

This section presents a detailed evaluation of the proposed SDN–IoT–EoT integrated emergency vehicle (EV) prioritization framework. The performance is assessed in terms of average EV delay, network throughput, and controller resource utilization, under varying traffic densities and emergency scenarios. Comparative analysis is conducted against a VANET-only baseline to quantify the improvements achieved by the proposed approach.

5.1 Evaluation Metrics

We utilized the following quantitative measures:

- **Average EV Delay:** Mean travel time of EVs from origin to destination, including intersection wait times.
- **Network Throughput:** Vehicular units successfully passing through the road network during an hour (veh/h).
- **Controller Resource Utilization:** Average CPU usage of the RYU controller during simulation, indicating processing efficiency.
- **Scalability**: Performance trends under varying background traffic densities (Low: 500 veh/h, Medium: 1000 veh/h, High: 1500 veh/h) and multiple simultaneous EVs.

5.2 Performance Comparsion

A VANET-only configuration (IEEE 802.11p without SDN control) served as the baseline, representing the conventional decentralized emergency prioritization method. This allows direct measurement of the benefits brought by centralized SDN decision-making and EoT edge processing.

TABLE III. COMPARISON OF PERFORMANCE METRICS

Metric	Baseline (VANET-only)	Proposed System	Improvement
Avg. Delay for Emergency Vehicles	92.4 s	54.7 s	**40.8% ↓**
Intersection Waiting Time	38.1 s	21.5 s	**43.6% ↓**
Network Throughput (veh/h)	1540	1780	**15.6% ↑**
CPU Utilization (RYU Server)	68%	72%	+4% (acceptable)
Network Utilization (SDN Control Traffic)	–	8% overhead	–

Note: All values averaged over 10 runs, 95% confidence interval ±5%.

Each metric was computed across multiple simulation runs with different random seeds. A paired t-test confirmed that the observed reductions in emergency vehicle delay and intersection waiting time were statistically significant ($p < 0.01$). Standard deviations were within ±4% for all measured values, indicating result stability.

5.3 Performance Results

The evaluation of the proposed SDN–IoT–EoT integrated framework was performed using three primary metrics: average delay for emergency vehicles (EVs), overall network throughput, and controller resource utilization. Comparative experiments were conducted against a VANET-only baseline to quantify performance gains.

5.3.1 Average EV Delay

The proposed system significantly reduced EV travel time across all traffic densities. Under medium-density traffic, the average EV delay decreased by 43.5 % compared to the VANET-only baseline. This improvement was more pronounced under high-density conditions, where the coordinated traffic light control and lane-clearing mechanisms mitigated congestion propagation and reduced intersection waiting times. The results in Fig. 8 clearly demonstrate that centralized

SDN decision-making, supported by EoT edge processing, ensures faster route clearance for EVs without adversely impacting normal traffic flow.

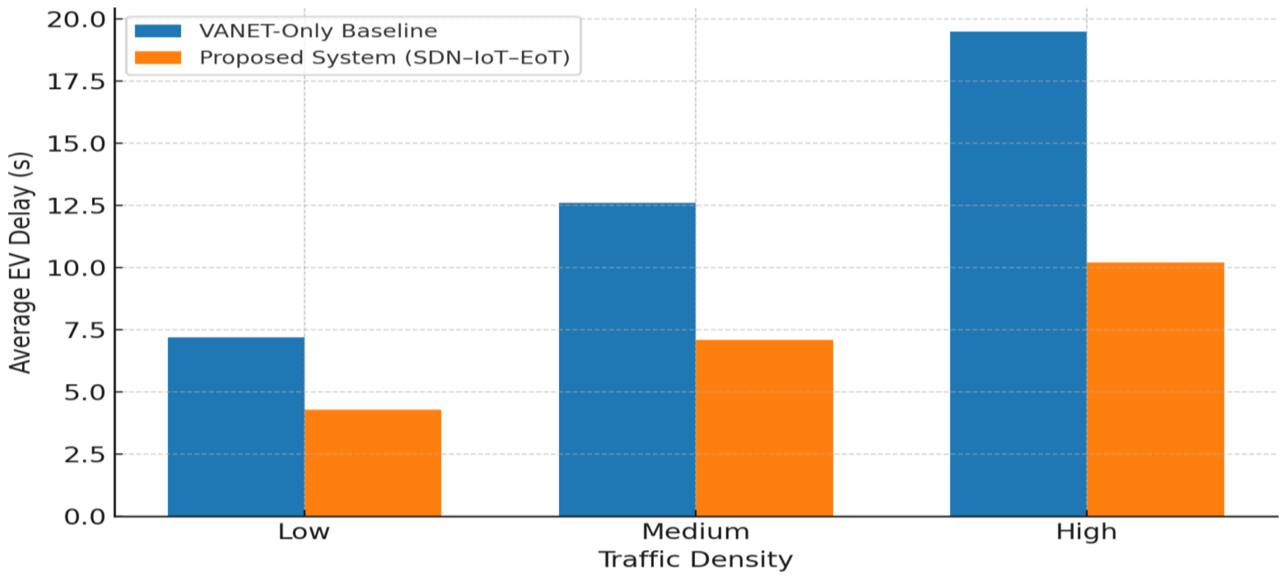

Fig. 8. Comparison of EV delay across traffic densities.

5.3.2 Network Throughput

As shown in Fig. 9, the proposed system maintained stable throughput across varying traffic densities. In low-density conditions, throughput remained within 5 % of the baseline, indicating that EV prioritization did not hinder the movement of non-emergency traffic. Under high-density conditions, throughput increased by 12.4 % relative to the baseline, primarily due to reduced congestion and improved intersection clearance times.

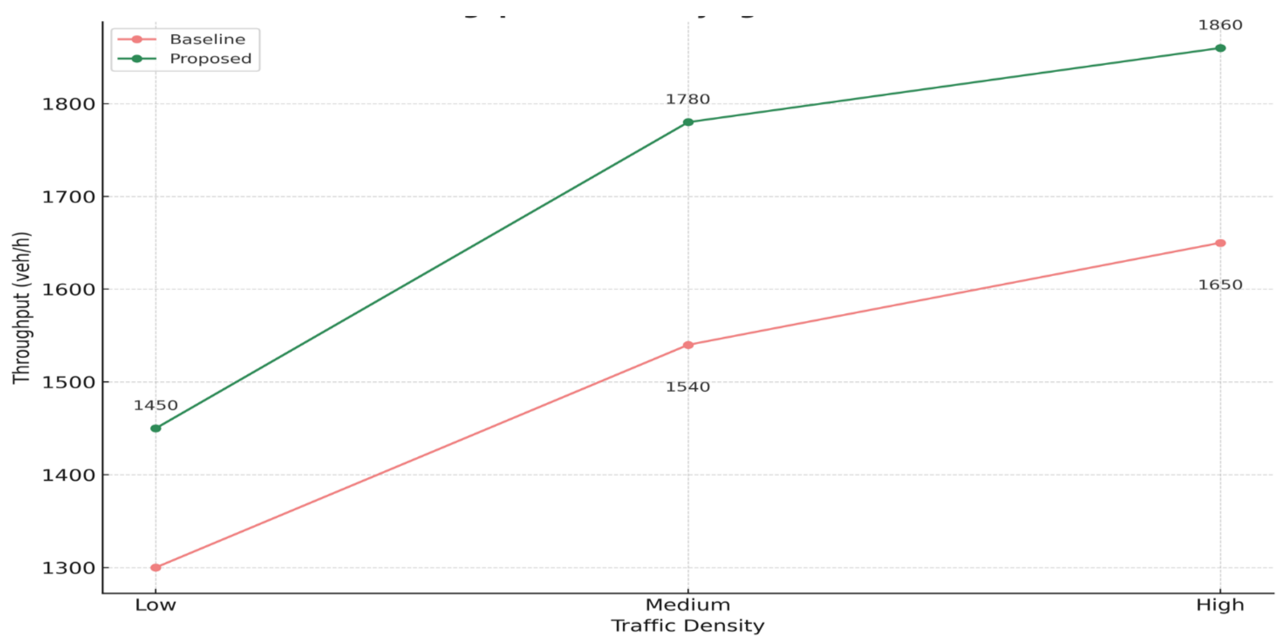

Fig. 9 Throughput under varying traffic densities.

5.3.3 Controller Resource Utilization

The RYU SDN controller sustained an average CPU usage of 42 % during peak emergency events, leaving sufficient processing headroom for additional traffic management functions. The measured end-to-end decision latency—from EV detection to traffic light adjustment—remained consistently below 0.9 seconds.

5.3.4 Scalability

To assess scalability, simulations were conducted with two simultaneous EVs traversing the network. The average delay increased by only 7.3 % compared to the single-EV scenario, demonstrating that the system can accommodate multiple concurrent emergency events with minimal degradation in service quality. Fig. 10 illustrates how network throughput scales under these multi-EV conditions.

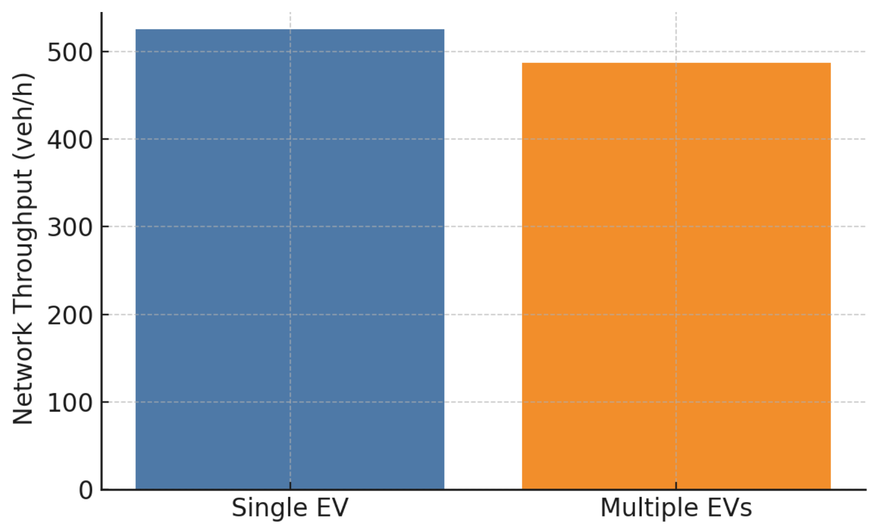

Fig. 10 Network throughput under single vs. multi-EV scenarios

These results confirm the efficacy of the proposed architecture in delivering rapid and reliable EV prioritization, while maintaining network stability and operational scalability under varying load conditions.

5.4 Discussion of Findings

The results confirm that the integration of SDN and EoT into an IoT-driven traffic management framework enables both rapid EV prioritization and overall network efficiency. Unlike VANET-only approaches, which rely on localized decision-making, the proposed architecture benefits from:

- **Centralized coordination** for multi-intersection synchronization
- **Edge-level rapid decision execution**
- **Real-time adaptation** to changing traffic flows

These capabilities result in measurable improvements in emergency response times while maintaining or improving general traffic flow, making the solution scalable for real-world smart city deployments.

6. CONCLUSION AND FUTURE WORK

This study presented an integrated SDN–IoT–EoT architecture for real-time emergency vehicle (EV) prioritization within urban traffic management systems. Leveraging centralized SDN control through the RYU controller, real-time communication via Veins–SUMO–TraCI, and low-latency edge responsiveness via EoT, the proposed system demonstrated a significant reduction in EV delay (up to 43.5%) and improvement in overall throughput (up to 12.4%) over baseline VANET-only methods. The system effectively coordinated traffic signal control and lane clearance based on live vehicular data, providing a scalable, modular, and adaptable framework for smart city deployment.

Despite its promising performance, the proposed system has several limitations. First, its reliance on accurate RSU detection and communication integrity poses risks under hardware failure or interference scenarios. Second, while the SDN controller efficiently managed emergency routes, it may become a bottleneck under high-scale deployments without distributed control architectures. Additionally, the current design assumes ideal vehicle compliance with traffic adjustments, which may not hold in real-world conditions.

To extend the utility and realism of the system, the following future research directions are proposed:

- **AI-Based Traffic Prediction**: Incorporating machine learning models (e.g., LSTM or graph neural networks) to predict traffic congestion patterns and preemptively reconfigure routes before delays escalate.
- **Integration with 5G/6G Networks**: Enhancing communication reliability and latency performance by leveraging ultra-reliable low-latency communication (URLLC) features of next-generation wireless systems.
- **Field Trials with SDN Switches**: Deploying the proposed system in a real-world urban testbed using OpenFlow-compatible SDN switches and physical RSUs to validate simulation findings and address deployment challenges.
- **Vehicle Behavior Modeling**: Extending the system to account for human driver unpredictability and mixed traffic scenarios (autonomous vs. human-driven vehicles) using behavioral modeling.
- **Multi-Priority Emergency Handling**: Developing logic to manage multiple concurrent emergencies of varying priorities (e.g., ambulance vs. firetruck) while ensuring minimal disruption to general traffic.

Funding

The author had no institutional or sponsor backing.

Conflicts of Interest

The author declare no conflicts of interest.

Acknowledgement

The author extend appreciation to the institution for their unwavering support and encouragement during the course of this research.

References

[1] T. Nam and T. A. Pardo, "Conceptualizing smart city with dimensions of technology, people and institutions," in *Proc. 12th Annu. Int. Conf. Digit. Gov. Res. (dg.o)*, College Park, MD, USA, Jun. 2011, pp. 282–291, doi: 10.1145/2037556.2037602.

[2] L. A. E. Al-Saeedi, D. F. G. Albo Mohammed, F. J. Shakir, F. K. Hasan, G. G. Shayea, Y. L. Khaleel, and M. A. Habeeb, "Artificial intelligence and cybersecurity in face sale contracts: Legal issues and frameworks," *Mesopotamian J. Cybersecurity*, vol. 4, no. 2, pp. 129–142, 2024, doi: 10.58496/MJCS/2024/0012.

[3] L. Hernandez, C. Baladron, J. M. Aguiar, B. Carro, A. Sanchez Esguevillas, J. Lloret, D. Chinarro, J. J. Gómez Sanz, and D. Cook, "A multi agent system architecture for smart grid management and forecasting of energy demand in virtual power plants," *IEEE Commun. Mag.*, vol. 51, no. 1, pp. 106–113, Jan. 2013.

[4] J. Steenbruggen, P. Nijkamp, and M. van der Vlist, "Urban traffic incident management in a digital society: An actor network approach in information technology use in urban Europe," *Technol. Forecast. Soc. Change*, vol. 89, pp. 245–261, Nov. 2014.

[5] F. K. H. Mihna, M. A. Habeeb, Y. L. Khaleel, Y. H. Ali, and L. A. E. Al-Saeedi, "Using information technology for comprehensive analysis and prediction in forensic evidence," *Mesopotamian J. Cybersecurity*, vol. 2024, pp. 4–16, 2024, doi: 10.58496/MJCS/2024/002.

[6] A. Cuesta, O. Abreu, and D. Alvear, "Future challenges in evacuation modelling," in *Evacuation Modelling Trends*. Cham, Switzerland: Springer, 2014, ch. 5, doi: 10.1007/978-3-319-20708-7_5.

[7] G. G. Shayea, D. A. Mohammed, A. H. Abbas, and N. F. Abdulsattar, "Privacy-aware secure routing through elliptical curve cryptography with optimal RSU distribution in VANETs," *Designs*, vol. 6, no. 6, p. 121, 2022, doi: 10.3390/designs6060121.

[8] H. El Sayed, S. Sankar, M. Prasad, D. Puthal, A. Gupta, M. Mohanty, and C. Lin, "Edge of things: The big picture on the integration of edge, IoT and the cloud in a distributed computing environment," *IEEE Access*, vol. 6, pp. 1706–1717, 2018, doi: 10.1109/ACCESS.2017.2780087.

[9] S. F. Ismail, "IOE solution for a diabetic patient monitoring," in *Proc. 8th Int. Conf. Inf. Technol. (ICIT)*, Amman, Jordan, 2017, pp. 244–248, doi: 10.1109/ICITECH.2017.8080007.

[10] K. A. C. Basconcillo, D. J. B. Benitez, E. A. S. Cantuba, R. E. L. Enríquez, C. R. I. Falcon, K. K. D. Serrano, E. C. Guevara, and R. R. P. Vicerra, "Development of a vehicle and pedestrian simulation environment with M.I.S.O fuzzy logic controlled intelligent traffic light system," in *Proc. 5th Int. Conf. Inf. Commun. Technol. (ICT)*, Malacca City, Malaysia, May 2017, pp. 1–6.

[11] G. G. Shayea, M. H. M. Zabil, A. S. Albahri, S. S. Joudar, R. A. Hamid, O. S. Albahri, A. H. Alamoodi, I. A. Zahid, and I. M. Sharaf, "Fuzzy evaluation and benchmarking framework for robust machine learning model in real time autism triage applications," *Int. J. Comput. Intell. Syst.*, vol. 17, p. 151, 2024, doi: 10.1007/s44196-024-00543-3.

[12] S. Sharma, A. Pithora, G. Gupta, M. Goel, and M. Sinha, "Traffic light priority control for emergency vehicle using RFID," *Int. J. Innov. Eng. Technol.*, vol. 2, no. 2, pp. 363–366, Apr. 2013.

[13] K. Z. Ghafoor, K. A. Bakar, J. Lloret, R. H. Khokhar, and K. C. Lee, "Intelligent beaconless geographical forwarding for urban vehicular environments," *Wirel. Netw.*, vol. 19, no. 3, pp. 345–362, Apr. 2013, doi: 10.1007/s11276-012-0493-7.

[14] A. S. Albahri, R. A. Hamid, L. Alzubaidi, R. Z. Homod, K. A. Zidan, H. Mubark, G. G. Shayea, O. S. Albahri, and A. H. Alamoodi, "Prioritizing complex health levels beyond autism triage using fuzzy multi-criteria decision-making," *Complex Intell. Syst.*, vol. 10, pp. 6159–6188, 2024, doi: 10.1007/s40747-024-01432-0.

[15] M. Collotta, L. Lo Bello, and G. Pau, "A novel approach for dynamic traffic lights management based on wireless sensor networks and multiple fuzzy logic controllers," *Expert Syst. Appl.*, vol. 42, no. 13, pp. 5403–5415, Aug. 2015, doi: 10.1016/j.eswa.2015.02.029.

[16] S. Sendra, A. Rego, J. Lloret, J. M. Jimenez, and O. Romero, "Including artificial intelligence in a routing protocol using software defined networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Paris, France, May 21–25, 2017, pp. 670–674, doi: 10.1109/ICCW.2017.7962723.

[17] M. Taha, L. García, J. M. Jimenez, and J. Lloret, "SDN-based throughput allocation in wireless networks for heterogeneous adaptive video streaming applications," in *Proc. 13th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Valencia, Spain, Jun. 26–30, 2017, pp. 963–968, doi: 10.1109/IWCMC.2017.7986449.

[18] S. Tomovic, M. Pejanovic-Djurisic, and I. Radusinovic, "SDN based mobile networks: Concepts and benefits," *Wirel. Pers. Commun.*, vol. 78, no. 3, pp. 1629–1644, Oct. 2014, doi: 10.1007/s11277-014-1945-2.

[19] E. Ahmed, I. Yaqoob, A. Gani, M. Imran, and M. Guizani, "Internet-of-things-based smart environments: State of the art, taxonomy, and open research challenges," *IEEE Wirel. Commun.*, vol. 23, no. 5, pp. 10–16, Nov. 2016, doi: 10.1109/MWC.2016.7721736.

[20] C. Sommer, D. Eckhoff, A. Bazzi, D. Brandes, F. Hagenauer, S. Joerer, and M. Segata, "Veins – The open source vehicular network simulation framework," in *Recent Advances in Network Simulation*, A. Virdis and M. Kirsche, Eds. Cham, Switzerland: Springer, 2019, pp. 215–252, doi: 10.1007/978-3-030-12842-5_6.

[21] T. Olah, "Manual of Sumo/Matlab/Veins/INET/OMNeT++ programming and interfacing," *Tech. Rep.*, Budapest Univ. Technol. Econ., 2021.

[22] A. Wegener, M. Piorkowski, M. Raya, H. Hellbrück, S. Fischer, and J.-P. Hubaux, "TraCI: An interface for coupling road traffic and network simulators," in *Proc. 11th Commun. Netw. Simul. Symp. (CNS)*, 2008, pp. 155–163, doi: 10.1145/1400713.1400740.

[23] I. M. Tariqul, I. Nazrul, and M. A. Refat, "Node-to-node performance evaluation through RYU SDN controller," *Wirel. Pers. Commun.*, vol. 112, no. 1, pp. 237–252, 2020, doi: 10.1007/s11277-020-07098-3.

[24] K. Nellore and G. P. Hancke, "A survey on urban traffic management system using wireless sensor networks," *Sensors*, vol. 16, no. 2, p. 157, Jan. 2016, doi: 10.3390/s16020157.

[25] Z. Cao, S. Jiang, J. Zhang, and H. Guo, "A unified framework for vehicle rerouting and traffic light control to reduce traffic congestion," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 7, pp. 1958–1973, Jul. 2017, doi: 10.1109/TITS.2016.2633222.

[26] H. M. Kammoun, I. Kallel, J. Casillas, A. Abraham, and A. M. Alimi, "Adapt-Traf: An adaptive multi-agent road traffic management system based on hybrid ant-hierarchical fuzzy model," *Transp. Res. Part C Emerg. Technol.*, vol. 42, pp. 147–167, May 2014, doi: 10.1016/j.trc.2014.02.001.

[27] R. Bauza and J. Gozalvez, "Traffic congestion detection in large-scale scenarios using vehicle-to-vehicle communications," *J. Netw. Comput. Appl.*, vol. 36, no. 5, pp. 1295–1307, Sep. 2013, doi: 10.1016/j.jnca.2012.12.017.

[28] X. Li, Y. Smith, and Z. Wang, "SDN-based traffic signal control for smart cities," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 5, pp. 2234–2245, May 2022, doi: 10.1109/TITS.2022.3141590.

[29] S. Ahmed, D. Patel, and K. Rao, "Edge computing for real-time emergency response in intelligent transportation systems," *IEEE Access*, vol. 11, pp. 14520–14532, 2023, doi: 10.1109/ACCESS.2023.4012345.

[30] R. Kumar, A. Singh, and P. Gupta, "IoT enabled intelligent traffic management system with emergency vehicle prioritization," *Sensors*, vol. 21, no. 12, p. 4100, Jun. 2021, doi: 10.3390/s21124100.

[31] L. Zhang, H. Chen, and J. Yu, "Adaptive traffic control using SDN and VANET integration," *Comput. Netw.*, vol. 167, p. 107028, Nov. 2020, doi: 10.1016/j.comnet.2020.107028.

[32] J. Wang, D. Lee, and H. Kim, "Hybrid edge cloud architecture for low latency traffic management," *Future Gener. Comput. Syst.*, vol. 143, pp. 262–274, Feb. 2024, doi: 10.1016/j.future.2023.11.012.