The Indonesian Journal of Computer Science
www.ijcs.net
Volume 14, Issue 3, June 2025
https://doi.org/10.33022/ijcs.v14i3.4845

Fusion-based Intelligent Congestion Management Algorithm for On-Road Traffic in Smart Cities

Ndivhuho Tshilongamulenzhe[1], Deon P. Du Plessi[2], Topside E. Mathonsi [3]
ndivhuhots@gmail.com[1], danieldp@uj.ac.za[2], mathonsite@tut.ac.za[3]
[1,3] Tshwane University of Technology
[2] University of Johannesburg

Article Information

Received : 16 Apr 2025
Revised : 30 Apr 2025
Accepted : 9 Jun 2025

Keywords

IEEE 802.11 standard, SNs, WSNs, FICM algorithm, NS-3, FAR, MTTD, DR, Traffic congestion.

Abstract

The concept of Wireless Sensor Networks (WSNs) has garnered significant global attention due to their wide range of applications. According to the IEEE 802.11 standard, WSNs are wireless networks consisting of sensor nodes (SNs) that interconnect via wireless communication links. These SNs are capable of sensing, processing, and wirelessly transmitting data, even in challenging environments. WSNs are primarily utilized for communication in various domains, including smart cities, healthcare, residential areas, and military applications. However, the deployment of WSNs in environments such as smart cities come with some challenges, particularly traffic congestion. Traffic congestion in smart cities, particularly during peak hours, is often caused by a high volume of vehicles traveling at the same time, resulting in delays, accidents, and inefficiencies under various weather conditions, including sunny and rainy days. Therefore, this paper proposes a novel algorithm called the Fusion-based Intelligent Congestion Management (FICM) algorithm, developed through the integration of the Navigation Reference Spatial Data (NRSD) algorithm and Fusion-based Multimodal Abnormal Detection (FMAD) algorithm. The objective of FICM is to mitigate on-road traffic congestion within smart cities effectively. The algorithm's performance was evaluated using Network Simulator 3 (NS-3) by comparing its effectiveness with the NRSD and FMAD algorithms. Under sunny weather conditions, the NS-3 simulation results revealed that the FICM algorithm achieved an average False Alarm Rate (FAR) of 0.83%, a Mean Time to Detection (MTTD) of 76.0%, and a Detection Rate (DR) of 84.3%, outperforming both the NRSD and FMAD algorithms. Similarly, under rainy weather conditions, the FICM algorithm demonstrated an average FAR of 14.09%, an MTTD of 57.03%, and a DR of 78.04%, surpassing the performance of the NRSD and FMAD algorithms within the smart city environment.

A. Introduction

Wireless Sensor Networks (WSNs) have emerged as a revolutionary data collection technique that is used on a daily basis worldwide. WSNs are frequently used to improve network infrastructure in smart cities, healthcare, home automation, industrial monitoring, and military applications [1, 2]. WSNs consist of sensor nodes (SNs), which are used to collect data from various environments and transmit it to a designated location, such as a base station (BS) or other specified location [3, 4]. These SNs are primarily powered by batteries, which are difficult to recharge once depleted. As a result, WSNs often encounter energy consumption challenges [5]. SNs can sense, process, and transmit data over wireless channels in complex environments. They transmit data from a source node to a destination node, which the data is typically stored in the cloud. Furthermore, network users can then access this data using various devices [6, 7].

This paper demonstrates the implementation of WSNs in a smart city to monitor on-road traffic congestion caused by a high volume of vehicles traveling across various lanes. When a significant number of vehicles move through a specific area of the smart city, the SNs collect data by detecting traffic levels on the road. Therefore, the collected data is then transmitted to the BS and subsequently to the cloud for analysis. Additionally, traffic congestion in a smart city, which is caused by a high volume of vehicles traveling across various lanes, increases the risk of accidents. Then, during the peak hours, heavy movement of vehicles normally leads to time delays, noise pollution, and wasted fuel, contributing to the increased of air pollution. This traffic congestion negatively impacts the environment by causing air and noise pollution, while also reducing the quality of life due to longer travel times and higher fuel consumption. The time wasted in traffic impacts productivity and had negative consequences on the economy and society. In smart cities, there are limitations based on traffic congestion which include parking space, traffic jams, illnesses due to the frustration and stress of the drivers, blocking lanes caused by accidents, and unexpected spikes in traffic [8-10]. Author [1,3, 8-10] further demonstrated that traffic congestion has become one of the main challenges in smart cities.

Hence, the proposes algorithm will set the priorities based on the collected data by SNs to mitigate traffic congestion across various lanes in a smart city. A notification is then sent to all drivers within the city's radius, informing them about traffic jams and suggesting alternative, congestion-free routes. Then, this will enable drivers to avoid areas with traffic congestion within the smart city. The input parameters used in this paper include traffic volume, traffic density, occupancy, traffic congestion index, and travel time, all of which are utilized to monitor and predict traffic congestion using SNs in a smart city.

Then, a Fusion-based Intelligent Congestion Management (FICM) algorithm will be developed by integrating the Navigation Reference Spatial Data (NRSD) algorithm proposed by [11] and Fusion-based Multimodal Abnormal Detection (FMAD) algorithm proposed by [12] to reduce traffic congestion caused by the increased number of vehicles within smart cities. The NRSD algorithm which was proposed by [11] demonstrated that their proposed algorithm managed to collect traffic data and route the traffic on available routes to alleviate traffic congestion which causes air and noise pollution in smart cities. The FMAD algorithm proposed by [12] demonstrated that it can detect possible traffic congestion before it occurs based on the traffic features such as speed, occupancy, and flow. Their proposed FMAD algorithm achieves this process because it first cleans and pre-processes the traffic data that was collected in the areas within the smart cities. Thereafter, it computes the absolute of the first derivative of each sample in the traffic feature to determine its anomaly score. Hence, the FICM algorithm proposed by this paper will use a real-time process when collecting traffic data and route the traffic on available routes as demonstrated by the NRSD algorithm and the detection of possible traffic congestion as demonstrated by the FMAD algorithm.

The remainder of this paper is structured as follows: In Section II, this paper presents the wireless sensor network overview. The overview of traffic congestion in WSNs is presented in Section III. In Section IV, the techniques used for traffic congestion in WSNs is presented. The overview of integrated algorithms is presented in Section IV. The related work is discussed in Section V. In Section VI, the proposed system architecture is presented. In Section VII, the design of fusion-based intelligent congestion management algorithm is presented. The proposed FICM algorithm is presented in Section VIII. In Section IX, the Implementation of the proposed algorithm is presented. The simulation results are presented in Section X. In Section XI, a conclusion is presented, and Section XII covers the acknowledgement for this research project.

B. Wireless Sensor Netwrok Overview

The WSNs are wireless systems composed of spatially distributed, dedicated sensors designed to monitor and record environmental conditions, transmitting the collected data to a central location, as illustrated in Figure 1 [13, 14]. WSNs can measure environmental conditions such as traffic density, sound levels, noise pollution, humidity, temperature, and many other factors, depending on the specific functionality of the sensors deployed in a given environment [15]. Furthermore, these sensors can transmit data from the SNs, which are responsible for collecting information from smart city roads. These SNs are interconnected with Road-Side Units (RSUs), transceivers mounted on roads or pedestrian pathways, and On-Board Units (OBUs) installed in vehicles for traffic monitoring. The data is then forwarded to central devices, such as sink nodes and BS, which receive and process the collected information (see Figure 1):

Figure 1: Smart city WSNs structure.

Fig 1 shows that SNs are strategically placed near smart city roads with vehicle movement, enabling the sensors to collect data effectively [16, 17]. These SNs are capable of sensing, collecting, and transmitting data from one node to another. The sink node then receives the data from the SNs and forwards it to the internet, where it is stored on a management node, such as a server. Users can subsequently access the data via the internet. All SNs utilize a shared communication channel (transmission channel) managed through multiple access protocols. These include random access protocols, which allow a transmitting node to use the channel at its full capacity, and media access control (MAC) protocols, which coordinate access among active nodes.

C. Result and Discussion

WSNs are prone to traffic congestion, where unpredicted events such as packet and node congestion occur within the network [18-21]. The traffic congestion primarily happens near the sink node when packets are transmitted through a wireless link from SNs to the BSs or sink node [22]. Furthermore, traffic congestion also occurs on the link level during packet transmission within WSNs [22, 23]. Therefore, due to the resource constraints of WSNs, there are several deployed nodes that are event-driven in WSNs. Hence, WSNs in some applications like event-driven applications, face significant challenges due to a huge amount of transmitted data from the source nodes to the destination nodes [24, 25]. This normally causes traffic congestion which leads to a breakdown of a network and decreases the network throughput and energy efficiency, increase of packet loss, delays, and reduction of network lifetime [26]. There are several types of WSNs traffic congestion, which is based on how packets are lost, namely: link-level congestion and node-level congestion (buffer overflow).

(a) Link-level congestion: is congestion that occurs when multiple SNs transmit packets using the same transmission channel which become overloaded within the network (see Figure 2.) [27]. Therefore, when packet congestion occurs, those transmitted packets may fail to reach the destination node because the packets overload the transmission channel. This type of packet congestion increases the packet service time and decreases the link utilization within the network.

Figure 2: Link-Level Congestion.

(b) Node- level congestion: is congestion that occurs at the specific node which receives the transmitted packets from different SNs and leads to buffer overflow [28] (see Figure 3). Furthermore, buffer overflow is normally caused by the mismatch of the packet departure rate and arrival rate during packet transmission or when the packet arrival rate is higher than the packet service time. When this buffer overflow occurs, all the incoming packets are simply dropped, which leads to increased queuing delays and packet loss.

Figure 3. Node-Level Congestion.

In Figure 2 and Figure 3, this paper shows the two possible ways that cause packet congestion within WSNs. This paper will focus on node-level congestion. To minimize and manage the traffic congestion that occurs in WSNs within the smart city, an appropriate algorithm should be developed. Doing so, will allow the collected packets within the area to reach their destination nodes which will lead to less traffic congestion within the smart city. Hence, the FICM algorithm is developed to minimize the traffic congestion in WSNs within the smart city. Furthermore, the load balancing in this research study will be implemented to avoid any link-node congestion during the packet distribution so that all packets can reach their intended destination nodes in time.

D. Techniques used for Traffic Congestion in WSNs

This There are various traffic congestion techniques used in WSNs which normally concentrate on assigning fair and efficient transmission data rates to each node within the network [29]. These techniques adjust the traffic rate based on packer service time along with fair packet scheduling algorithms. These traffic congestion techniques are divided into three parts: congestion notification, congestion detection, and congestion mitigation.

(a) Congestion notification

Congestion notification is a mechanism used for reporting the congestion state of the packets, and node up until the sink or Gateway [24, 30]. This process of notification helps in monitoring, detecting, and signaling the presence of traffic congestion when it occurs within a network. Moreover, once congestion is recognized in a network, the congestion notification will employ the congestion control technologies, namely: adjusting transmission rates and rerouting traffic to regulate the flow of packets and alleviate congestion [31, 32]. There are some common techniques used for congestion notification in WSNs, namely:

- Queue Length Monitoring: this mechanism monitors the length of the traffic packets that queue at links or at the nodes which can provide some insights into the congestion [33]. When the queue length of the traffic packets increases beyond a certain threshold, it then shows that there is potential for packet congestion. Then, the nodes will notify other nodes about the congestion status within the network [34].
- Traffic Monitoring: traffic monitoring observes the traffic patterns in the network [35]. These traffic patterns help in identifying areas with high packet transmission rates, which may lead to congestion. The congestion notification can then notify the possibilities of congestion based on the observed traffic conditions using the patterns [36].
- Packet Loss Monitoring: packet loss monitoring refers to the monitoring of packet loss rate which can be the indirect indication of any possible congestion within a network [37]. In a network, if packet congestion occurs, there is a high increase in packet loss. Then the nodes will detect this increase and notify other nodes or the sink about the congestion.

(b) Congestion detection

Congestion detection is another mechanism used to identify and monitor congested packets within the network [38]. The mechanism was developed to maintain the overall network performance and reliability of packet transportation from the source node to the destination node. There are some common techniques used for congestion detection in WSNs, namely:

- Energy Considerations: The SNs in WSNs are often powered by batteries with limited energy [39]. Hence, congestion detection processes need to consider the energy constraints of nodes, because it improves the reliability of the nodes used in WSNs.
- Adaptive Routing Protocols: The congestion detection aspect is also used in adaptive routing protocols within the WSNs [38]. This process is implemented because these protocols allow congestion detection to dynamically adjust the routes that packets take through the entire network based on the current conditions during packet transmission. This helps the traffic packets to be rerouted away from the congested areas, which distributes the traffic more evenly [40].
- Efficient congestion detection: The efficient congestion detection aspect within the WSNs was developed to maintain the reliability, scalability, and longevity of the entire network, especially in applications where timely and accurate data is critical [41]. Hence, this aspect was developed because most of the mechanisms often focus on the development of lightweight and energy-efficient which is suitable for resource-constrained sensor nodes [42].

(c) Congestion Mitigation

Congestion mitigation is another mechanism which involves the strategies and techniques used to alleviate the congestion problems in the network [42]. This plays a crucial role in ensuring the network operates reliably and efficiently. There are some common congestion mitigation techniques used in WSNs, namely:

- Traffic Prioritization: The congestion mitigation normally prioritizes certain types of packets and the critical nodes within the WSNs [24, 43]. Moreover, this technique also assists in giving the packets priority to ensure that crucial information is transmitted with lower latency, even in congested situations.
- Cross-Layer Optimization: Cross-layer optimization is also implemented in congestion mitigation [44]. This technique is used when the information from multiple layers of the communication protocol stack is used to make decisions.
- Dynamic Channel Allocation: Dynamic channel allocation is implemented when there are multiple communication channels available [42]. Then, dynamic channel allocation techniques can be employed to switch channels based on the congestion level. This technique helps in distributing the communication load across different channels and reducing the congestion.

The traffic congestion mechanisms discussed above demonstrated how traffic congestion is being minimized in WSNs. Therefore, paper does consider these mechanisms during the development of the proposed FICM algorithm. By augmenting the capabilities of the FICM algorithm implementation, this paper endeavors to achieve significant reductions in traffic congestion levels across the smart city infrastructure.

E. Related work

This Over the decades, a lot of work has been conducted in this area and many researchers have contributed to the field of WSNs within the smart city environment and provided their solutions.

The Optimal Restricted Driving Zone (ORDZ) which uses the Genetic Algorithm was proposed by [45] to select suitable controlled traffic zones that can optimally control traffic congestion and air pollution will result in improved citizen satisfaction within the smart cities. The authors further demonstrated that the proposed ORDZ uses an augmented genetic algorithm and determinant theory to randomly generate different foursquare zones within the smart cities. The fitness function in ORDZ considers a trade-off between traffic load and citizen satisfaction. The NS-2 simulation tool was used to simulate the proposed ORDZ. The simulation results demonstrated that the ORDZ managed to control traffic congestion by 30.6% when compared with other methods. However, the proposed ORDZ does not consider the traffic patterns when there are more vehicles within the smart cities, which leads to traffic congestion.

An Intelligent Traffic Management Systems (ITMS) was proposed by [46] to predict the optimum routes based on vehicle categorization, accident occurrences, and levels of precipitation. The authors further demonstrated that the proposed ITMS makes use of the green corridor concept to allow the emergency services to travel without facing any kind of traffic congestion in smart cities. NS-2 was used to test the effectiveness of the proposed ITMS. The simulation results demonstrated that the proposed ITMS manages to predict the optimum routers for emergency services within the smart cities when compared with the fuzzy sliding mode congestion control algorithm (FSMC) and inter-vehicle communication. However, the proposed ITMS does not consider the traffic mobilization patterns when there is a high volume of vehicles travelling within the cities, which leads to traffic density and an increase of travelling time.

The Optimized Weight Elman Neural Network (OWENN) algorithm that makes use of efficient Internet of Things (IoT)-based traffic prediction was proposed by [47] to monitor the traffic jams caused by the increase of vehicles and accidents that occur in the smart cities worldwide. The authors stated that the proposed OWENN algorithm consists of five different phases, namely: IoT data collection, feature extraction, classification, optimized traffic IoT values, and traffic signal control system. Moreover, the OWENN algorithm enhances the IoT values by using the controlled Intel 80,286a when processing the data within the smart cities. The NS-2 simulation tool was used to carrier out the simulation results of the proposed OWENN algorithm. Their simulation results demonstrated that the proposed OWENN algorithm managed to reduce the traffic jams in smart cities with 91.23% accuracy. However, the proposed OWENN algorithm does not consider the delay of data during the transmission within the network, which makes the traffic congestion persist within the smart city.

The Low-Cost Locations Discovery (LCLD) Scheme was proposed by [48] to reduce the localization error of the vehicle movement and the cost of vehicle broadcasting based on mobile vehicles and unmanned aerial vehicles (UAVs) in a smart city. The authors illustrated that they used the Large Error Rejection (LER) algorithm and the UAV Same Position Broadcast Repeat (USPBR) algorithm to minimize the localization error and reduce the cost of broadcasting vehicles within the smart city. The authors stated that they used the Optimized Network Engineering Tools (OpNet) as their simulation tool to test the effectiveness of the proposed LCLD scheme. Their simulation results demonstrated that the proposed LCLD scheme managed to reduce the localization error by 78.8% and the Cost reductions was reduced by up to 16.5% when compared with the Adaptive UAV Flight Path Planning (AUPPP) algorithm and USPBR algorithm. However, the proposed LCLD scheme does not consider the time delays of data and traffic congestion index that has been distributed by sensors within the smart city. This remains as a problem which cause traffic congestion within the smart city. This research study proposes the use of PBCC algorithm to reduce the traffic congestion in smart city.

The Priority-based Adaptive Traffic Signal Control (PATSC) system was proposed by [49] to manage the traffic congestion that occurs at the road intersections. The authors stated that the proposed PATSC system acts according to the context of the traffic status within the intersections. Furthermore, the PATSC system makes use of a selection method based on the lane priority. The lane priority is calculated based on traffic parameters: vehicle density and waiting time. The MATLAB simulation tool was used to test the effectiveness of the proposed PATSC system. The simulation results demonstrated that the traffic flow has increased by 31.05% when compared with the fixed-time traffic signal system in the traffic light. However, the proposed PATSC system does not consider the travel time and fuel consumption when sensors collect data within the road intersections, which leads to traffic congestion within the smart city.

A new Adaptive Traffic Light Control System (ATLCS) algorithm was proposed by [50] to assist traffic management authorities in efficiently reducing traffic congestion in smart cities. The authors illustrated that to reduce traffic congestion in the smart cities, the proposed ATLCS algorithm will synchronize the number of traffic lights controlling consecutive junctions by creating a delay between the times at which each of them switches to green at a given direction based on the load of vehicles in that area. They further stated that the dynamic updated ATLCS algorithm is implemented based on the number of vehicles waiting at each junction. This will minimize the number of occurrences of the 'stop and go' phenomenon that caused by traffic congestion within the smart city. They use the Network Simulator 2 (NS-2) simulation tool for the performance evaluation of their proposed ATLCS. Their simulation results demonstrated that the average travel time of vehicles traveling in the synchronized direction has been significantly reduced by up to 39% when compared with the non-synchronized fixed-time Traffic Light Control Systems. However, the proposed ATLCS algorithm does not consider the emergence vehicles which are on the other direction of the road.

The use of Emergency Vehicle Priority System (EVPS) was proposed by [51] to determine the priority level of an Emergency Vehicles (EV's) and estimate the number of necessary signal interventions while taking into account the effect of those interventions on the flow of traffic in the roadways next to the EV's travel path depending on the kind and severity of an incident. The authors further demonstrated that the proposed EVPS estimates the number of green signal interventions to attain the quickest incident response while concomitantly reducing the impact on other vehicles within the smart cities. The OpNet simulation tool was used to simulate the proposed EVPS. The simulation results demonstrated that the EVPS managed to produce an appropriate number of interventions that reduce emergency response time significantly. However, the proposed EVPS does not consider the traffic density when reducing the response time, which makes traffic congestion persist within the smart city. Table 1 below shows the summary of the related work.

Table 1. Summary of related work

Authors	**Proposed Algorithms / methods**	**Advantages**	**Limitations**
Jan et al., [45]	ORDZ	it selects suitable controlled traffic zones that can optimally control traffic congestion	it does not consider the traffic patterns when there are more vehicles.

		and air pollution.	
Khanna et al., [46]	ITMS	it uses of the green corridor concept for emergency services.	the traffic mobilization patterns were not considered when there is a high volume of vehicles.
Neelakandan et al., [47]	OWENN	it enhances the IoT values by using the controlled Intel 80,286a when processing the data	the transmission delay of data was not taken to consideration.
Amalfitano et al., [48]	LCLD Scheme	the LER and USPBR algorithms used to minimize the localization error and reduce the cost of broadcasting vehicles	does not consider the time delays of data and traffic congestion index that has been distributed by sensors.
Mondal et al., [49]	PATSC	make use of a selection method based on the road priorities, and acts according to the context of the traffic status within the intersections.	does not consider the travel time and fuel consumption when sensors collect data within the road intersections.
Aleko et al., [50]	ATLCS	it synchronize the number of traffic lights based on time creating a delay within the consecutive junctions.	the emergency vehicles which are on the other direction of the road were not considered.
Karmakar et al., [51]	EVPS	estimates the number of green signal interventions to attain the quickest incident response while concomitantly reducing the impact on other vehicles	traffic density is not taken to consideration when reducing the response time.

F. Proposed System Architecture

The proposed architectural design represents the integrated WSNs infrastructure in smart cities. The system is designed to facilitate the seamless movement of road users by providing current updates on real-time based on traffic congestion in the city. This connectivity is established by wireless technologies such as RSUs, OBUs, BSs, and the Internet, forming a robust wireless network. In this network, vehicles are equipped with OBUs that allow them to act as dynamic nodes. This enables real-time communication between vehicles and network infrastructure, including RSUs and BSs. As a result, road users can receive an immediate update on traffic conditions, allowing them to make informed decisions to avoid congestion.

The Internet, represented as a cloud, shows the use of cloud computing services to process, store, and disseminate large amounts of data generated by WSN. This facilitates data management and analysis and incorporates artificial intelligence (AI) to predict traffic patterns and suggest optimal routes. However, the operation of the Global Positioning System (GPS) is essential to the system's ability to provide traffic updates that are specific to the location. By ensuring that all nodes, infrastructure, vehicles, even traffic lights, are interconnected, the system supports GPS functions to update and inform in real time about the condition of the city's roads. The proposed system aims to improve traffic flow, reduce congestion, and make urban transport experiences more efficient.

Figure 4. Proposed WSNs System Architecture

As shown in Figure 4, BSs are the central nodes that manage network communication. They serve as access points to the Internet and are connected to various sensors and RSUs within their range. They collect, process, and relay information to and from other nodes. RSUs serve as local communication hubs. They are typically connected to BSs and communicate with passing vehicles via short-range wireless communication technologies such as dedicated short-range communications (DSRC) or cellular networks. OBUs are installed in vehicles and allow them to communicate with RSUs and other vehicles through vehicle-to-infrastructure (V2I) or vehicle-to-vehicle (V2V) communication. OBUs can also connect to passenger personal devices, allowing them to interact with the smart city network. Various sensors are deployed throughout the city to monitor conditions. These can include traffic flow sensors, weather sensors, pollution sensors, and more. They are wirelessly connected to the nearest RSU. The cloud acts as the backbone for data storage, processing, and analytics. Then data is being received from the RSUs and provides the computational power needed for complex tasks such as traffic prediction and system-wide optimisation.

G. Design of Fusion-based Inteligent Congestion Management Algorithm

This paper assimilated the existing solutions algorithms such as NRSD and FMAD algorithms.

(a) NRSD algorithm

The proposed FICM algorithm integrates NRSD and FMAD algorithms to overcome traffic congestion within smart city, leading to traffic density, time delay, and air and noise pollution. The proposed FICM algorithm uses NRSD algorithm to estimate and classify the congestion on the road within the smart city. The NRSD algorithm makes use of Machine Learning (ML) techniques namely: predictive analytics to predict future trends and events based on historical data, for example, predicting traffic congestion on roads, anomaly detection to identify abnormal patterns or events, and optimisation algorithm to optimise resource allocation and decision-making processes when estimating and classifying the traffic congestion on roads from different lanes within the smart city. Furthermore, this technique implemented by the NRSD algorithm can analyse, identify, and collect the pattern based on the traffic congestion that occurs on the road. This process also allows the road user to be alerted of any traffic congestion that may occur on a certain lane (lane 1, lane 2, and lane 3) within the smart city. The NRSD algorithm provides innovative services to enable drivers to choose alternative routes with no or less traffic congestion to avoid any delays caused by traffic jams and robot malfunctions. This paper applies the technique used by NRSD algorithm to estimate possible on-road traffic congestion before occurring so that road users (drivers) may be aware of possible delays on certain lanes within the smart city. The traffic congestion may be calculated using Equations (1) and (2).

$$T_P = \frac{1}{2}\sum_{k=1}^{v_i} -Pi\left(1 + \overline{P_e^J}\right) \tag{1}$$

where T_P represents the estimated total number of vehicles within the city, v_i represent the set of vehicles in motion within the city, and Pi represents the vehicles which are in stationary. Meanwhile $\overline{P_e^J}$ represents the possibilities of the traffic congestion within the city.

$$C_n = T_P + 1\left(D_P * \left(\frac{1}{t_s - 1}\right)\right) + \left(1 + \frac{1}{t_s}\right) * 0.2 \quad (2)$$

where C_n represents the classified number of vehicles detected in the incoming lanes within the city, and D_P represents the delay caused by the incoming vehicles, while t_s represents the time per second taken by the incoming vehicles. Furthermore, the technique implemented to estimate the possible traffic congestion in the city before occurring can be defined by Equations (3) and (4).

$$ML_{ac} = Pi + 0.1\left(\frac{C_n}{t_s - 1}\right) \quad (3)$$

$$ML_{ac} = \left(\frac{C_n}{t_s - 1} + Pi\right) * \frac{0.1}{t_s + 1} \quad (4)$$

where ML_{ac} represents the machine learning technique that monitors the movement of vehicles within the city. Later, this paper will show the integration of all the algorithms used to design the proposed FICM algorithm.

(b) FMAD algorithm

The proposed FICM algorithm is designed and implemented to detect traffic congestion based on the features such as speed, occupancy, and traffic behaviour within the city. The FMAD algorithm detects traffic congestion in real-time without any human intervention or prior knowledge. This algorithm uses three performance metrics to detect traffic congestion, namely: Detection Rate (DR), False Alarm Rate (FAR), and Mean Time to Detection (MTTD). Furthermore, the FMAD algorithm uses the average speed to calculate the speed of vehicles that are moving from one place to another within the smart city area.

When the FMAD algorithm detects traffic congestion in the smart city, it redirects the road users to other roads so that there is an increase in flexibility and stability of vehicles movement within the smart city. Therefore, this paper implements the FMAD algorithm to provide detection of traffic congestion and redirect road users to alternative routes within the smart city, so there is high flexibility and stability of vehicles movement within the smart city. The imbalance of vehicle movement within the smart city area causes an increase in time delay, congestion rate, accidents, and air and noise pollution. Therefore, the sequence of traffic congestion observation based on speed and occupancy is calculated using Equation (5).

$$X = (X_1, X_2, \dots, X_n) \quad (5)$$

where X represents the time series of the length based on the given traffic feature, n represents the observation sequence, and $(1,2, \dots n)$ represents the denoted time sequence when the observations based on the speed and occupancy were recorded by the sensors within the smart city. The average time of the vehicle's speed is calculated using Equation (6).

$$T_t = a + X(1-a) * T_{t-1} \tag{6}$$

where T_t represents the average time calculated using the vehicle's speed within the smart city. To determine the absolute amount of the change in traffic congestion using vehicle speed or occupancy over time is calculated using Equation (7).

$$AA_c = \begin{cases} |T_{t-1} - \overline{T_t}| & t = [1,2] \\ \left|\frac{1}{4}(T_{t+1} - T_{t-2})\right| & t \in [3 \ldots n-2] \\ |T_{t-1} - T_t + 1| & t = [n-1, n] \end{cases} \tag{7}$$

where AA_c represents the absolute amount of the change in traffic behaviour in terms of increasing or decreasing within the smart city. When a significant change in traffic behaviour occurs (for example, when the traffic speed goes from low to high or vice versa), the AA_c value changes to indicate the potential of traffic congestion. The real-time congestion detection for short-term and long-term averages for AA_c is calculated using Equations (8), (9), and (10).

$$\tilde{\mu} = \sum_{i=0}^{\tilde{w}-1} \left(AA_c + \frac{t_{t-i}}{\tilde{w}}\right) \tag{8}$$

$$\mu = \sum_{i=0}^{w-1} \left(AA_c + \frac{t_{t-i}}{w}\right) \tag{9}$$

$$\sigma = \frac{0.2}{t_t}\left(\frac{\tilde{\mu} + \mu}{AA_c}\right) * 100 \tag{10}$$

where $\tilde{\mu}$ represents the short-term average of AA_c, and μ represents the long-term average of AA_c, while the σ represents the standard deviation in real-time congestion detection. The $\tilde{w}$ and w represents the sliding window protocol for short-term and long-term respectively. The percentage of the detected traffic congestion based on the performance metrics is calculated using Equations (11) and (12).

$$DR = \frac{No.of\ congestions\ detected}{No.of\ true\ congestions} * 100 \tag{11}$$

where DR represents the Detection Rate percentage of the correct detected traffic congestions in the network.

$$FAR = \frac{No.of\ false\ alarms\ signals}{No.of\ non-congestion\ instance} * 100 \tag{12}$$

where FAR represents the False Alarm Rate percentage signal to the total number of non-incident instances. While the Mean Time to Detection is calculated using Equation (13).

$$MTTD = \frac{\sum_{i=1}^{N}(t_c - t_r)}{N} * 0.01 \quad (13)$$

where N represents the number of detected congestions, and t_c represents the alarm time when the congestion was detected, while t_r represents the time when the congestion appeared on the given road segment. Then, the MTTD demonstrated the average of the congestion detection delay at the different times when the congestion was detected by the algorithm and when it appeared.

H. Proposed Algorithm

The proposed FICM algorithm was developed through the integration of NRSD and FMAD algorithms. The proposed FICM algorithm will minimise the congestion rate and time delay caused by traffic congestion. Furthermore, this paper aims to reduce air and noise pollution, traffic density, and longer travel time within the smart city as given by algorithm 1. Additionally, some of the threshold values considered in the design of the proposed algorithm include travel time, queue length, environmental factors, and traffic density.

Algorithm 1. Fusion-based Intelligent Congestion Management (FICM) algorithm

INPUT: Data packets from RSU, OBUs and Sensors
OUTPUT: Traffic Congestion Analysis and Management Actions:
1. Initialization:
2. **Do**
3. Define parameters for RSUs, OBUs, Sensors, and constants used for traffic analysis
4. Establish encrypted transmission channels for secure data transfer
5. **WHILE** all nodes in the city lanes are initialized
6. Data collection and routing process:
7. **DO WHILE** there are data to be collected and routed:
8. **IF** error in data transmission **Then**:
9. Apply the error correction mechanism and retry the transmission
10. **ELSE**
11. Compress data packets and transmit them using efficient batching
12. **LOOP**
13. Traffic Congestion Detection using NRSD:
14. **FOR EACH** vehicle in the smart city **DO:**
15. Estimate the total number of vehicles (T_P) using equation (1)
16. Classify the number of vehicles detected in the incoming lanes (C_n) using

equation (2)
17. Apply machine learning techniques for predictive analytics and anomaly detection using equations (3) and (4) to monitor vehicles movement and predict congestion
18. **LOOP UNTIL** traffic analysis is complete for all vehicles
19. Traffic Congestion Management using FMAD:
20. **DO WHILE** managing detected traffic congestion
21. Utilize the sequence of traffic congestion observation based on speed and occupancy using equation (5)
22. Calculate the average time of the vehicle's speed (T_t) using Equation (6)
23. Determine the absolute amount of change in traffic congestion using vehicle speed or occupancy over time with Equation (7)
24. Calculate the real-time congestion detection for short-term and long-term average for $\tilde{\mu}$ using equation (8), (9), and (10)
25. **IF** real-time data indicates high congestion **THEN:**
26. Apply predictive analysis for pre-emptive traffic flow adjustments
27. Execute real-time feedback actions based on traffic signal adjustments and alternative routes
28. Calculate the percentage of the detected traffic congestion in different lanes using equations (11), (12), and (13).
29. **LOOP**
30. Continuous Monitoring and Evaluation:
31. **DO**
32. Monitor system performance against the established key performance indicator
33. Schedule regular system updates and maintenance for optimal operation
34. **WHILE** the system is operational
35. **END**

The proposed FICM algorithm utilizes real-time data from the environment and can adapts to different traffic patterns within the smart city by incorporating various sources, such as traffic sensors, GPS data, and historical traffic trends.

I. Implementation of the Proposed Algorithm

The FICM algorithm was implemented using the Satellite protocol and IEEE 802.15.4 (Zigbee) model in Network Simulator 3 (NS-3) version 35. NS-3 is a versatile network simulator that forms the basic simulations that run virtual machines on the Linux operating system. On the simulation platform, this paper illustrates the NAM simulation scenario based on the topology that is created for the simulations as shown in Figure 5. This paper implemented 6 BSs within the NS-3 simulator in order to gather information within the smart city and send it to the cloud. The vehicles are equipped with the OBUs which allows them to act as relay nodes, and thus able to communicate and send the messages (packets) to the RSUs.

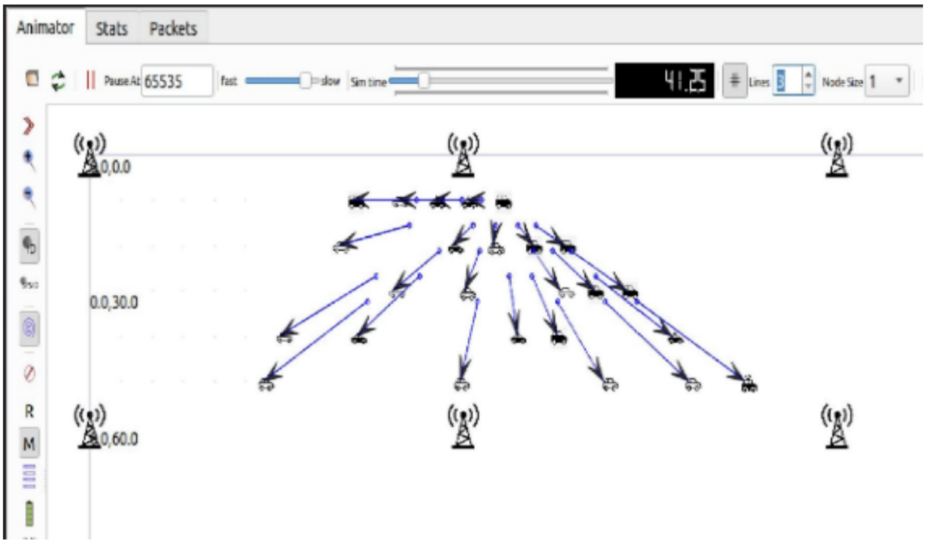

Figure 5. Smart city NAM Simulation Scenario 1.

The topology illustrated in Figure 5 represents the smart city WSN architecture, used to carry out the simulations. In these simulations, the network topology covered an area of 900m x 900m, with 6 fixed RSUs in the smart city and 25 placed vehicles equipped with the OBUs for communication. The vehicles communicate with each other using the OBUs and with the RSUs through the same technology. This setup is used to compare the three algorithms, as exemplified in Figure 5 and figure 6.

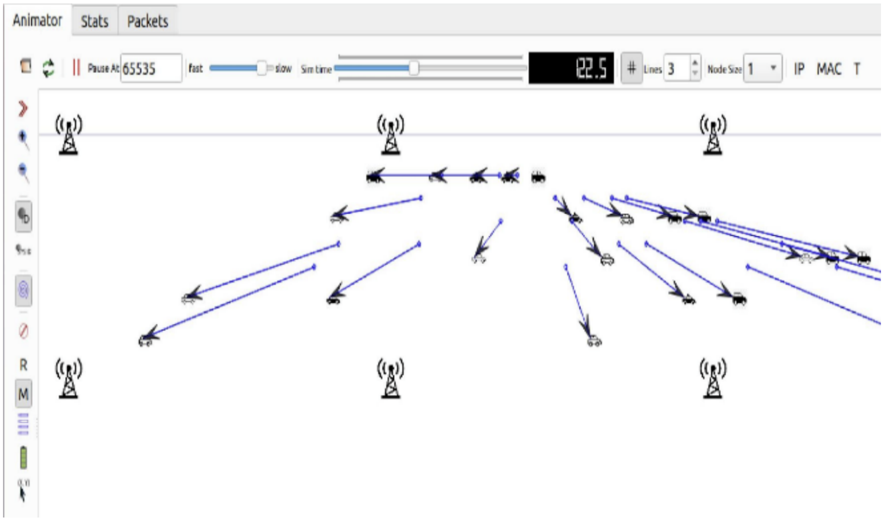

Figure 6. Smart city NAM Simulation Scenario 2

The above Figure 6 illustrates the communication between the sensor nodes and the RSUs within the smart city. The OBUs send information to the nearest RSUs, which then forward the information to the cloud. As vehicles continue to move, the OBUs communicate with the nearest RSUs, and the RSUs will then share the information about the on-road traffic congestion with the driver so that they can be alerted about any related traffic across different lanes within the smart city. This process occurs because, during the simulation configuration the RSUs were set up as nodes to broadcast the information from the OBUs to the vehicles, whether vehicles are stationary or in motion. As vehicles continue to move along the lanes, the next RSUs gather information about any related traffic from the OBUs.

This paper further utilised the dataset structure presented on Table 2. The dataset provides a detailed view of traffic data within the smart city and demonstrated the features of dataset containing over 1.2 million records. The dataset can be found on the link provided within the Footnote1. This paper used the *futuristic_city_trafic* dataset. Each record represents a unique snapshot of various features related to traffic conditions within the smart city. Before training the model, this paper ensures that the dataset undergoes several preprocessing steps to enhance data quality and optimize performance. The data cleaning process includes handling missing values through imputation and removing duplicate or inconsistent entries was performed. Afterward, the dataset is split into training, validation, and test sets, often using stratified sampling to preserve class distribution. The proposed FICM algorithm utilizes 80% of the dataset for training and 20% for testing.

[1] https://drive.google.com/file/d/1G8VJ_P-9p7yWZC9mg7OzhPt6J98MS2Ql/view?usp=sharing

Table 2. Dataset structure features

Vehicle type	Day of Week	# Hour of Day	# Speed	# Is Peak Hour	Energy Consumption of vehicle	Traffic Density
Truck	Sunday	20	29.4268	0	14.7134	0.5241
Bus	Wednesday	2	118.8	0	143.5682	0.3208
Autonomous Vehicle	Wednesday	16	100.3904	0	91.264	0.0415
Truck	Thursday	8	76.8	1	46.0753	0.1811
Autonomous Vehicle	Saturday	16	45.2176	0	40.1934	0.4544
Autonomous Vehicle	Thursday	20	30.5179	0	37.5562	0.0843
Autonomous Vehicle	Monday	21	43.9222	0	39.042	0.0293
Truck	Sunday	20	29.4268	0	14.7134	0.5241
Bus	Wednesday	2	118.8	0	143.5682	0.3208

J. Simulation Result

This paper discusses the performance of the proposed FICM algorithm based on the findings analysed during simulations in terms of the three-performance metrics: DR, FAR, and MTTD.

1. DR: occurs when the ratio of true positive and the total nonself samples are identified by the detector set, where true positive and false negative are the tallies of true positive and false negative.
2. FAR: is the number of false positives relative to the sum of the number of true positives and the false positives, it further measures how often a detected target is correctly identified, offering a way to evaluate the accuracy of detection systems in areas like sensing and traffic monitoring.
3. MTTD: is used to calculate the average amount of time it takes to detect a traffic or anomaly within the system after it occurs. To measure the time taken, the total amount of time it takes to detect the traffic during a given period is added and divide it by the number of traffic detected.

This paper compared the FICM algorithm with the NRSD and FMAD mechanisms. This paper chose the NRSD, mainly because the algorithm managed to collect traffic data and route the traffic on available routes to alleviate traffic congestion in smart cities which cause air and noise pollution. The FMAD algorithm was chosen because it can detect the possible traffic congestion before it occurs based on traffic features such as speed, occupancy, and flow. Both the NRSD and FMAD use multiple parameters to on-road traffic congestion within the smart city. The review of the literature claims that both the NRSD and the FMAD work best when it comes to mitigating the on-road traffic congestion within the smart city. Therefore, this paper attempted to ascertain whether the proposed FICM algorithm could outperform those two algorithms.

1. Average Detection rate

The DR occurs when the ratio of true positive and the total nonself samples are identified by the detector set, where true positive and false negative are the tallies of true positive and false negative. Therefore, it is important that this paper improves the detection rate in order to improve traffic congestion within the smart city. The proposed FICM algorithm was compared with the NRSD and the FMAD algorithms to test the effectiveness of the proposed algorithm, as illustrated in Figure 7.

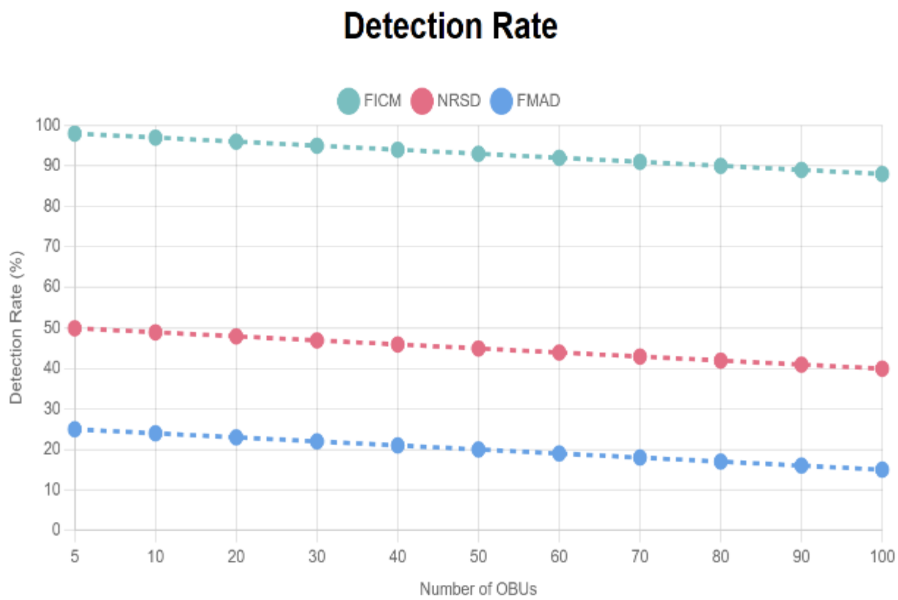

Figure 7. Average Detection Rate.

The simulation results showed that the proposed FICM algorithm, NRSD and FMAD each experienced different detection rates of 84.03%, 46.02%, and 18.02%, respectively. The proposed FICM managed to increase the detection rate by 56%. The overall experimental evaluations that the proposed FICM algorithm generally outperforms both the NRSD and FMAD algorithms in terms of detection rate. Estimating the total number of vehicles and analysing traffic behaviour, including trends of increase or decrease, within the smart city which contributed to its improved performance.

2. Average False Alarm Rate

Normally, the FAR is based on the number of false positives relative to the sum of the number of true positives and the false positives, the FAR further measures how often a detected target is correctly identified. Therefore, it is important for this paper to reduce the FAR within the smart city. The experimental evaluations for FAR are illustrated in Fig 8. This is to determine the effectiveness of the proposed FICM algorithm compared with the NRSD and the FMAD algorithms.

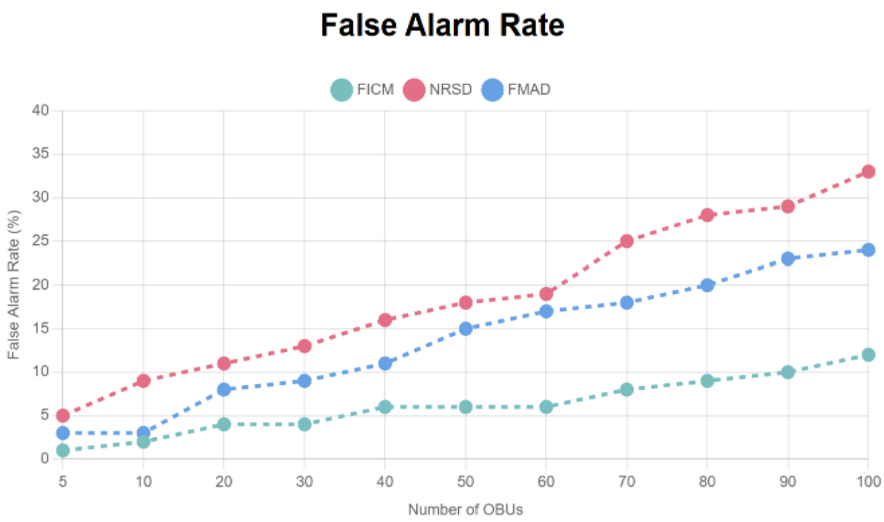

Figure 8. Average False Alarm Rate.

The proposed FICM algorithm, NRSD and FMAD, each obtained an average FAR of 0.83%, 2.08%, and 5.07%, respectively (see Figure 8). The proposed FICM algorithm produces less FAR, as it is reduced by 0.6% compared with the others. As can be seen from the experiment, the proposed FICM consistently outperforms the NRSD and FMAD algorithms throughout the entire simulation. The reason behind this superior performance is attributed to the consideration of traffic congestion changes based on vehicle speed and the critical importance of data transfer between OBUs and RSUs within the smart city.

3. Average Mean Time to Detection

In general, the MTTD is used to calculate the average amount of time it takes to detect a traffic or anomaly within the system after it occurs. This section discusses the evaluation and analysis of the effectiveness of the MTTD based on the proposed FICM, the NRSD and the FMAD algorithms. The proposed FICM algorithm produces better results for MTTD, as illustrated in Figure 9.

Figure 9. Average Mean Time to Detection.

The proposed FICM algorithm, when compared with the NRSD and FMAD, each obtained the MTTD average by 76.03%, 42.06 %, and 37.04%, respectively (See Figure 5.3). The FICM algorithm produced better MTTD average results, demonstrating that it does detect a traffic in the smart city. These results obtained through the integrated of the three algorithms as discussed in the literature.

K. Conclusion

The implementation of WSNs within the smart cities has grown rapidly. This is because WSNs are wireless systems composed of spatially distributed, dedicated sensors designed to monitor and record environmental conditions and transmit the collected data to a central location. WSNs are frequently used to improve network infrastructure in smart cities, healthcare, home automation, industrial monitoring, and military applications. As a result, this paper developed an algorithm that mitigates the on-road traffic congestion caused by a high volume of vehicles traveling across various lanes. The proposed FICM algorithm shows the improvement on: DR, FAR, and MTTD during the simulation results while compared with the NRSD and FMAD algorithms. This paper did not consider the issue of security in WSNs. Therefore, other researchers can work on ensuring that FICM algorithm deals with data integrity and security.

L. Acknowledgment

The authors would like to thank Tshwane University of Technology for financial support. The authors declare that there is no conflict of interest regarding the publication of this paper.

M. References

[1] S. Daousis, N. Peladarinos, V. Cheimaras, P. Papageorgas, D. D. Piromalis, and R. A. Munteanu, "Overview of Protocols and Standards for Wireless Sensor Networks in Critical Infrastructures," *Future Internet,* vol. 16, no. 1, p. 33, 2024.

[2] T. M. Tshilongamulenzhe, T. E. Mathonsi, and D. DuPlessis, "A Survey of Intelligent Traffic Control Algorithm in Wireless Sensor Networks," in *2022 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2022: IEEE, pp. 1156-1160.

[3] G. Kumar, R. S. Rathore, K. Thakur, A. Almadhor, S. A. A. Biabani, and S. Chander, "Dynamic routing approach for enhancing source location privacy in wireless sensor networks," *Wireless Networks,* vol. 29, no. 6, pp. 2591-2607, 2023.

[4] T. M. Tshilongamulenzhe, T. E. Mathonsi, D. P. Du Plessis, and M. I. Mphahlele, "Intelligent Traffic Routing Algorithm for Wireless Sensor Networks in Agricultural Environment," *Journal of Advances in Information Technology,* vol. 14, no. 1, pp. 46-55, 2023.

[5] O. S. Egwuche, A. Singh, A. E. Ezugwu, J. Greeff, M. O. Olusanya, and L. Abualigah, "Machine learning for coverage optimization in wireless sensor networks: a comprehensive review," *Annals of Operations Research,* pp. 1-67, 2023.

[6] A. Bagwari *et al.*, "An Enhanced Energy Optimization Model for Industrial Wireless Sensor Networks Using Machine Learning," *IEEE Access,* 2023.
[7] T. Mazhar *et al.*, "Quality of service (QoS) performance analysis in a traffic engineering model for next-generation wireless sensor networks," *Symmetry,* vol. 15, no. 2, p. 513, 2023.
[8] S. Tekkesinoglu, A. Habibovic, and L. Kunze, "Advancing Explainable Autonomous Vehicle Systems: A Comprehensive Review and Research Roadmap," *arXiv preprint arXiv:2404.00019,* 2024.
[9] Z. Huang and B. P. Loo, "Urban traffic congestion in twelve large metropolitan cities: A thematic analysis of local news contents, 2009–2018," *International journal of sustainable transportation,* vol. 17, no. 6, pp. 592-614, 2023.
[10] H. M. Rai, A. Pal, S. Mishra, and K. K. Shukla, "Use of Internet of Things in the context of execution of smart city applications: a review," *Discover Internet of Things,* vol. 3, no. 1, p. 8, 2023.
[11] R. K. Bharti, D. Suganthi, S. Abirami, R. A. Kumar, B. Gayathri, and S. Kayathri, "Optimal Extreme Learning Machine based Traffic Congestion Control System in Vehicular Network," in *2022 6th International Conference on Electronics, Communication and Aerospace Technology*, 2022: IEEE, pp. 597-603.
[12] M. Bawaneh and V. Simon, "Novel traffic congestion detection algorithms for smart city applications," *Concurrency and Computation: Practice and Experience,* vol. 35, no. 5, p. e7563, 2023.
[13] M. Toloueiashtian, M. Golsorkhtabaramiri, and S. Y. B. Rad, "An improved whale optimization algorithm solving the point coverage problem in wireless sensor networks," *Telecommunication Systems,* vol. 79, no. 3, pp. 417-436, 2022.
[14] P. Vijayalakshmi, M. Saravanan, M. T. Rani, M. Ashok, R. Palaniappan, and V. Nagaraj, "A Novel Hybridclustering Model Forwireless Sensor Networks," in *2023 International Conference on Artificial Intelligence and Knowledge Discovery in Concurrent Engineering (ICECONF)*, 2023: IEEE, pp. 1-8.
[15] A. K. Raghuwanshi and V. S. Chaudhary, "Energy Efficient Medium Access Control (MAC) Layer Protocol for Wireless Sensor Network," *SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology,* vol. 14, no. 03, pp. 270-279, 2022.
[16] D. d. F. Medeiros, C. P. d. Souza, F. B. S. d. Carvalho, and W. T. A. Lopes, "Energy-Saving Routing Protocols for Smart Cities," *Energies,* vol. 15, no. 19, p. 7382, 2022.
[17] A. Banerjee *et al.*, "Building of efficient communication system in smart city using wireless sensor network through hybrid optimization technique," *AI and IoT for Smart City Applications,* pp. 15-30, 2022.
[18] G. Sahar, K. A. Bakar, S. Rahim, N. A. K. K. Khani, and T. Bibi, "Recent advancement of data-driven models in wireless sensor networks: a survey," *Technologies,* vol. 9, no. 4, p. 76, 2021.
[19] S. Suma and B. Harsoor, "Detection of malicious activity for mobile nodes to avoid congestion in Wireless Sensor Network," in *2022 IEEE fourth*

international conference on advances in electronics, computers and communications (ICAECC), 2022: IEEE, pp. 1-6.

[20] N. Bagais, E. Fadel, and A. Al-Mansour, "An Efficient Path Planning Strategy in Mobile Sink Wireless Sensor Networks," *Computers, Materials & Continua,* vol. 73, no. 1, 2022.

[21] D. Wang, Y. Yi, S. Yan, N. Wan, and J. Zhao, "A node trust evaluation method of vehicle-road-cloud collaborative system based on federated learning," *Ad Hoc Networks,* vol. 138, p. 103013, 2023.

[22] S. I. Saheb, K. U. R. Khan, and C. S. Bindu, "A Hybrid Modified Ant Colony Optimization-Particle Swarm Optimization Algorithm for Optimal Node Positioning and Routing in Wireless Sensor Networks," *International journal of electrical and computer engineering systems,* vol. 13, no. 7, pp. 515-523, 2022.

[23] S. L. Yadav, R. Ujjwal, S. Kumar, O. Kaiwartya, M. Kumar, and P. K. Kashyap, "Traffic and energy aware optimization for congestion control in next generation wireless sensor networks," *Journal of Sensors,* vol. 2021, pp. 1-16, 2021.

[24] A. Grover, R. M. Kumar, M. Angurala, M. Singh, A. Sheetal, and R. Maheswar, "Rate aware congestion control mechanism for wireless sensor networks," *Alexandria Engineering Journal,* vol. 61, no. 6, pp. 4765-4777, 2022.

[25] H. Mohammad, K. Krishna, C. Gangadhar, and R. Mohammed, "Token Bucket Algorithm with Modernization Techniques to avoid Congestion in DEC Protocol of WSN," *International Journal of Electronics and Telecommunications,* pp. 507-513-507-513, 2023.

[26] M. Arieth, K. Anuradha, B. Harika, D. N. Ali, and S. Chowdhury, "Congestion Management of GSTEB Routing Protocol Using K-Means Algorithm in Wireless Sensor Network," *ECS Transactions,* vol. 107, no. 1, p. 13147, 2022.

[27] N. Yakymchuk, Y. Selepyna, M. Yevsiuk, S. Prystupa, and S. Moroz, "Monitoring of link-level congestion in telecommunication systems using information criteria," *Informatyka, automatyka, pomiary w gospodarce i ochronie środowiska,* vol. 12, no. 4, 2022.

[28] R. Zhang, J. Zhang, L. Liu, Z. Yan, and M. Dong, "Research on Active Detection Method of Network Congestion," in *GLOBECOM 2022-2022 IEEE Global Communications Conference*, 2022: IEEE, pp. 4685-4690.

[29] "futuristic_city_traffic Dataset." https://drive.google.com/file/d/1G8VJ_P-9p7yWZC9mg7OzhPt6J98MS2Ql/view?usp=sharing (accessed 20 October, 2024).

[30] J. P. A. León, L. J. de la Cruz Llopis, and F. J. Rico-Novella, "A machine learning based Distributed Congestion Control Protocol for multi-hop wireless networks," *Computer Networks,* vol. 231, p. 109813, 2023.

[31] S. Sunassee, A. Mungur, S. Armoogum, and S. Pudaruth, "A comprehensive review on congestion control techniques in networking," in *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, 2021: IEEE, pp. 305-312.

[32] T. Wang, H. Kan, Q. Sun, S. Xiao, and S. Wang, "Congestion Detection and Link Control via Feedback in RDMA Transmission," in *2022 International Conference on Service Science (ICSS)*, 2022: IEEE, pp. 1-4.

[33] Y. Su, P. Jiang, H. Chen, and X. Deng, "A QoS-Guaranteed and Congestion-Controlled SDN Routing Strategy for Smart Grid," *Applied Sciences,* vol. 12, no. 15, p. 7629, 2022.

[34] A. Navarro-Espinoza *et al.*, "Traffic flow prediction for smart traffic lights using machine learning algorithms," *Technologies,* vol. 10, no. 1, p. 5, 2022.

[35] F. Alzonem, G. Albrecht, D. Castellanos, M. Vandermeer, and B. Stansfield, "Ransomware detection using convolutional neural networks and isolation forests in network traffic patterns," 2024.

[36] H. Yahyaoui, M. F. Zhani, O. Bouachir, and M. Aloqaily, "On minimizing flow monitoring costs in large-scale software-defined network networks," *International Journal of Network Management,* vol. 33, no. 2, p. e2220, 2023.

[37] H. M. A. Fahmy, "Transport Protocols for WSNs," in *Concepts, Applications, Experimentation and Analysis of Wireless Sensor Networks*: Springer, 2023, pp. 311-343.

[38] J. Rocher-Gonzalez, J. Escudero-Sahuquillo, P. J. Garcia, and F. J. Quiles, "Congestion management in high-performance interconnection networks using adaptive routing notifications," *The Journal of Supercomputing,* vol. 79, no. 7, pp. 7804-7834, 2023.

[39] R. Nagaraju, S. Goyal, C. Verma, C. O. Safirescu, and T. C. Mihaltan, "Secure routing-based energy optimization for IOT application with heterogeneous wireless sensor networks," *Energies,* vol. 15, no. 13, p. 4777, 2022.

[40] E. Taheri, R. G. Kim, and M. Nikdast, "AdEle+: An Adaptive Congestion-and-Energy-Aware Elevator Selection for Partially Connected 3D Networks-on-Chip," *IEEE Transactions on Computers,* vol. 72, no. 8, pp. 2278-2292, 2023.

[41] P. Anitha, H. Vimala, and J. Shreyas, "Comprehensive review on congestion detection, alleviation, and control for IoT networks," *Journal of Network and Computer Applications,* p. 103749, 2023.

[42] S. Hudda, K. Haribabu, and R. Barnwal, "Energy efficient data communication for WSN based resource constrained IoT devices," *Internet of Things,* vol. 27, p. 101329, 2024.

[43] C. Trinh, B. Huynh, M. Bidaki, A. M. Rahmani, M. Hosseinzadeh, and M. Masdari, "Optimized fuzzy clustering using moth-flame optimization algorithm in wireless sensor networks," *Artificial Intelligence Review,* vol. 55, no. 3, pp. 1915-1945, 2022.

[44] J. Bai, T. Zhang, C. Wang, and G. Xie, "MA-CC: Cross-Layer Congestion Control via Multi-agent Reinforcement Learning," in *Science and Information Conference*, 2023: Springer, pp. 659-671.

[45] T. Jan, P. Azami, S. Iranmanesh, O. Ameri Sianaki, and S. Hajiebrahimi, "Determining the optimal restricted driving zone using genetic algorithm in a smart city," *Sensors,* vol. 20, no. 8, p. 2276, 2020.

[46] A. Khanna, R. Goyal, M. Verma, and D. Joshi, "Intelligent traffic management system for smart cities," in *Futuristic Trends in Network and Communication Technologies: First International Conference, FTNCT 2018, Solan, India, February 9–10, 2018, Revised Selected Papers 1*, 2019: Springer, pp. 152-164.

[47] S. Neelakandan, M. Berlin, S. Tripathi, V. B. Devi, I. Bhardwaj, and N. Arulkumar, "IoT-based traffic prediction and traffic signal control system for smart city," *Soft Computing,* vol. 25, no. 18, pp. 12241-12248, 2021.

[48] D. Amalfitano, M. Cutugno, U. Robustelli, and G. Pugliano, "Designing and Testing an IoT Low-Cost PPP-RTK Augmented GNSS Location Device," *Sensors,* vol. 24, no. 2, 2024.

[49] M. A. Mondal and Z. Rehena, "Priority-Based Adaptive Traffic Signal Control System for Smart Cities," *SN Computer Science,* vol. 3, no. 5, p. 417, 2022.

[50] D. R. Aleko and S. Djahel, "An efficient adaptive traffic light control system for urban road traffic congestion reduction in smart cities," *Information,* vol. 11, no. 2, p. 119, 2020.

[51] G. Karmakar, A. Chowdhury, J. Kamruzzaman, and I. Gondal, "A smart priority-based traffic control system for emergency vehicles," *IEEE Sensors Journal,* vol. 21, no. 14, pp. 15849-15858, 2020.