DEVELOPMENT OF AN INTELLIGENT TRAFFIC MANAGEMENT SYSTEM USING DYNAMIC TIME ALLOCATION TECHNOLOGY

BY

NWAOGWUGWU, NNANYEREUGO KELECHI CLINTON

B.ENG, M. ENG (FUTO)

REG NO: 20194200248

A PhD DISSERTATION

SUBMITTED TO THE POSTGRADUATE SCHOOL

FEDERAL UNIVERSITY OF TECHNOLOGY, OWERRI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF

DOCTOR OF PHILOSOPHY (PhD) DEGREE IN COMPUTER ENGINEERING

IN THE DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING,

SCHOOL OF ELECTRICAL SYSTEM ENGINEERING AND TECHNOLOGY

JUNE 2024

CERTIFICATION

I certify that 'Development of an Intelligent Traffic Management System Using Dynamic Time Allocation Technology' was carried out by Nwaogwugwu, Nnanyereugo K. C. (Reg No: 20194200248) in partial fulfillment of the requirements for the award of Doctor of Philosophy (PhD) degree, in the Department of Electrical and Electronic Engineering of the Federal University of Technology Owerri.

Engr. Prof. I. E. Achumba (Supervisor)	26/8/24 Date
Engr. Dr. C. K. Agubor (Co-Supervisor)	26/8/24 Date
Engr. Dr. O. C. Nosiri (Co-Supervisor)	26/08/24 Date
Engr. Dr. N. Chukwuchekwa (Head, Department of Electrical and Electronic Engineering)	26/8/24 Date
Engr. Prof. M. C. Ndinechi (Dean, School of Electrical Systems Engineering and Technology)	30/8/24 Date
Prof. B. O. Esonu (Dean, Post Graduate School)	Date
(External Examiner)	29/8/24 Date

DEDICATION

This dissertation is dedicated to Almighty God and to the Evergreen memories of my loving Wife, Julie, my ever caring, supportive Mum, Lady Flora and my academic-loving Dad, Chief Row.

ACKNOWLEDGEMENTS

My sincere thanks goes to Engr. Prof. M.C. Ndinechi, the present Dean, School of Electrical and Systems Engineering and Technology (SESET), FUTO, for his assistance and fatherly advice in the course of this programme. I must also thank immensely, Prof. B. Esonu (Dean, Post graduate School, FUTO) and Prof. C. O. Nweke (Associate Dean, Post Graduate School, FUTO), whose unassuming posture in piloting the affairs of the post graduate School, FUTO has in no small way aided the timely completion of this Ph.D work. I also want to express my indebtedness to Engr. Dr. N. Chukwuchekwa, the HoD, Electrical and Electronic Engineering, FUTO, for resilience and encouragements towards the completion of this programme on time.

Furthermore, I must acknowledge with great thanks, my Head Supervisor, Engr. Prof. (Mrs.) Ifeyinwa E. Achumba and her Co-Supervisors, Engr. Dr. Cosmos. K. Agubor, (Head, Electronics Engineering Department, FUTO), and Engr. Dr. Onyebuchi .C. Nosiri, (Director of ICT, FUTO), for both their technical and academic guidance; and for painstakingly going through my research work, giving all the necessary assistance. I must not fail to acknowledge Engr. Prof. Felix. K. Opara (my M.Eng Supervisor), for his mentorship and academic guidance throughout the duration of this research work.

I am also very grateful to Engr. Dr. Sam Okozi, the EEE Post graduate (PG) Coordinator, for all his relentless efforts towards the actualization of this programme. My gratitude further goes to Engr. Prof. G. A. Chukwudebe, Engr. Prof. E.N.C. Okafor, Prof. (Mrs) G.N.Ezeh, Engr. Prof. D.O. Dike, Engr. Prof. James Onojo, Engr. Dr. Isdore Akwukwaegbu, Engr. Dr. J.k. Obichere and Engr. Dr.Oliver Ozioko, for their respective unquantifiable efforts towards the realization of this research work. I also want to express my profound gratitude to Engr. Prof. S.O.E Ogbogu, (Professor Emeritus, EÈE Department, FUTO), whose advice and encouragements propelled me to Post graduate Studies in Federal University of Technology, Owerri. Thanks a lot Sir.

To my lovely Son, Kelechi Darren Nnanyereugo, daddy loves you and will always care for you. And to my amiable late wife, Julie Ihuaru Nnanyereugo, though death could not let you see this day which you worked and yawned for, but I still appreciate your love and support. Same Posthumous acknowledgement and gratitude goes to my dear parents, Hon. Chief (Sir) Rowland and Ezinne Lolo (Lady) Florence Nwaogwugwu, (All of blessed Memories), for their relentless Prayers, financial, moral and parental supports right from my cradle through my academic careers. I am and will ever be proud of you people. To you all my dear ones, I say, Requiescat In Pacem!

I am particularly grateful to my siblings, Dr.(Mrs.) Chioma Ike-Amadi, (Head, Department of Science Laboratory Technology, Abia State Polytechnic, Aba) and her husband, Hon. Justice Kenneth Ikechukwu Amadi, Ph.D, (Honourable Justice of the Court of Appeal, Federal Republic of Nigeria), Chief Barr. Ugochukwu Nwaogwugwu, (Traditional Prime Minister and The Enyi IV of Owuahia Eziudo Autonomous Community, Ezinihitte Mbaise LGA), Engr. Chikere Nwaogwugwu, His Worship, Ugbonma Favour Nwaogwugwu, (Magistrate, Imo State Judiciary, Owerri); my Nephews: Barr. Obinna Ike-Amadi, Master Ikechukwu Chidirim Ike-Amadi, (Osimiri ata-ata), Master Chigozirim Ugochukwu and Master Akachukwu Chikere-Nwaogwugwu; my little Nieces: Miss Somtochukwu Ike-Amadi, Miss Akuoma Ike-Amadi, Miss Tehila Ike-Amadi and Little Miss Amarachi Chikere-Nwaogwugwu. These people have contributed in no small measures towards the actualization of my ambition. In addition, I wish to acknowledge with thanks, the assistance of my Uncle, Late Nze Ephraim Ahaiwe, His wife, Lolo Magdalene Ahaiwe and his entire family, who have supported me to a great extent, financially, morally and otherwise.

Finally, to my Ph.D classmates, especially, Engr. Jeremiah Tashie Wosu, I want say a big 'Thank you' to you for all your unquantifiable assistance and academic support to me all through the course of this programme. To all my friends, colleagues, associates and well-wishers, worthy of note is Engr.(Pst.) James Chiemeke, my direct Boss/Supervisor in the office, who has being of immense spiritual, moral and financial assistance to me in difficult times of this programme. I am so grateful to you Sir. Also to Engr. (Sir) & Lady Nicholas Emeka Eneremadu, who ensured and facilitated my employment in Addax Petroleum Development Company of Nigeria Limited, Engr. & Mrs. Ignatius. A. Onumaegbu of Total E&P Nigeria Limited, Engr. (Sir) and Lady Uchenna Victor Eneremadu, Engr. Martin Chike Nwaogwugwu, Engr. Felix, Mr. Kachi and many others whose names were not mentioned here, I say thank you and God bless you all.

In conclusion, I thank God immensely for his enduring mercies, provisions and protections thus far. May his name alone be glorified.

Table of Contents

List of Tables

List of Figures

List of Plates

LIST OF ABBREVIATIONS

AI -Artificial Intelligence

ANFIS - Adaptive Neuro-Fuzzy Inference System

ANN – Artificial Neural Networks

APIs – Application Program Interface

CSGC - Copy Sensitive Graphical Codes

CSS – Cascading Style Sheets

DSRC - Dedicated Short Range Communications

DTAT - Dynamic Time Allocation Technology

FRSC - Federal Road Safety Corps

GPS - Global Positioning System

HTML – Hypertext Mark-up Language

HTTP – HyperText Transfer Protocol

IDE – Integrated Development Environment

IoT – Internet of Things

ITLS - Intelligent Traffic Light System

ITS - Intelligent Traffic Systems

IVC - Inter-Vehicle Communications

LIDAR - Light Detection and Ranging Sensor

LPR - License Plate Recognition

MANET - Mobile Adhoc Network

P2P - Peer-to-Peer

RFID - Radio Frequency Identification

RFID - Radio-Frequency Identification

RUP - Rational Unified Process

SDPs - Software Design Patterns

SLCML- Smart-Legal-Contract Markup Language

SNN - Simulated Neural Network

SQL – Structured Query Language

TCP/IP - Transmission Control Protocol/Internet Protocol

UI – User Interface

UML - Unified Model Language

URL – Uniform Resource Locator

USB – Universal Serial Bus

V2I - Vehicle-To-Infrastructure

V2V - Vehicle-To-Vehicle

VANET - Vehicular Ad Hoc Network

VDM-SL - Vienna Development Method-Specification Language

ABSTRACT

Considering the increase rate at which traffic congestion occur in urban cities which is associated with longer waiting time of vehicles on traffic queues as well as loss of productivity, fuel, time, fatigue and other health-threatening conditions; it is imperative to apply innovative, fully functional and affordable technologies to curb this challenge. Unfortunately, state-of-the-art traffic management systems are not able to solve this problem as they do not employ intelligent traffic control techniques at road junctions. This research developed an intelligent road traffic management system using Dynamic Time Allocation Technology (DTAT). In doing this, Modelio, an open-source Unified Model Language (UML) tool, was utilized in conjunction with object-oriented system analysis and design in analysing and modelling of typical traffic scenario at road junctions. A sensor network for the detection of vehicular presence and movement was designed using motion sensors and IP cameras. Furthermore, a camera system for capturing vehicle plate number of offending drivers, as well as traffic offence SMS gateway for communicating with offenders and appropriate authorities was developed. Fuzzy Logic and Artificial Neural Network (ANN) techniques as well as load balancing and remote procedural call (RPC) were applied in implementing the fundamental operations of the system. The system was simulated using Proteus 8 Professional in which a microcontroller was used to run the fuzzy logic operations, while the ANN runs on the data storage server and was used to analyse the patterns of stored data so that the system can learn traffic situations of the road with time. Results obtained indicate that the frequency at which traffic flows at a particular lane of the road within a period of time is a function of the number of vehicles that enter and leave the traffic zone at that point. Performance of the developed system shows that there is a balance in the flow of traffic for different lanes of the road intersecting at a point.

Keywords: *Road traffic, Fuzzy logic, ANN, Load balancer, RPC, Object-oriented system analysis and design, UML and DTAT.*

CHAPTER ONE
INTRODUCTION

1.1 Background of Study

Traffic congestion is a global phenomenon. A good number of major cities in the world experience heavy vehicular traffic. For example, according to Onyeneke (2018), many cities in Nigeria are faced with series of traffic congestions. Some cities in Nigeria face extreme traffic congestion where cars in the major cities travel at speeds of about 3-5km/hr. On most of the busy days of the week such as the workdays and during rush hours like the morning and evening hours, there exist extremely significant congestions popularly known as hold-ups or go-slow. Similarly, about 2,100 kilometers long roads around Hong Kong Island are serving over 762,000 vehicles. This has resulted to slow traffic flow around the country (Sin-Chun & Chok-Pang, 2020).

UKEssays (2018), revealed that causes of traffic congestions include – ‘car culture’, which means many people in the city or county own a private car, absence of underground railway, minimal use of public transport in cities, and weak public transportation infrastructure. Some other causes include lack of planning of city roads, narrow roads, growing population concentration, rapid urbanization, and increasing commercial activities.

Also, indiscipline on the part of vehicle drivers who oftentimes violate traffic rules contributes to high level of traffic congestion. Traffic rule violation also occurs when drivers do not obey the traffic light and when they drive against the traffic. Driving against the traffic is very pronounced in Nigeria and it has gotten several nick names like “driving one way,” etc. Furthermore, negligence on the part of constituted authorities, who are supposed to ensure that the roads are free from agents that could facilitate traffic congestion, as well as poorly implemented road traffic monitoring and control systems heavily contribute to road traffic congestions.

Road traffic congestions have many undesirable consequences including loss of productivity, resources (such as fuel), time, fatigue and other health-threatening conditions. In some cases, drivers tend to violate traffic rules and regulations due to tiredness which usually leads to accidents and consequently deaths (Dipti & Bongale, 2015; Stefan & Dirk, 2008). Naturally, increase in the average waiting time of vehicles on traffic queues potentially has health and economic implications. (Al-Sakran, 2015. Chowdhury, 2015; Bharani, 2018). Vehicles emit more Carbon (II) Oxide (CO) as they wait longer. This is poisonous to the human health and environment. Also, in health-emergency situations where a patient requires to be taken to the hospital for immediate medical attention, long traffic queues (due to longer average waiting time) could lead to losing the patient (Chowdhury, 2016).

To overcome these, there is an urgent need to improve Traffic Control systems (Ekedebe, 2013). Unfortunately, the current traffic control systems are mainly pre-defined fixed-cycle traffic light systems, which do not pay sufficient attention to the real traffic situations such as vehicle and pedestrian counts, delay and waiting time of the road users. Interestingly, Intelligent Traffic Control systems have come to the rescue.

Jahnavi, Prasanth, Priyanka, Sneheth, and Navya (2021) and Liang, Du, Wang and Han (2018) proved that intelligent real-time traffic control system should be an integral part of modern urban traffic control systems aimed at achieving optimal utilization of the road network. Sin-Chun and Chok-Pang (2020) defined Intelligent Traffic Systems and Services as the integration of information and communications technology with transport infrastructure, vehicles and users. To improve the traveling experiences and enjoy safety while on the road, (Sin-Chun & Chok-Pang, 2020) suggests that designing an intelligent traffic system (ITS) that fulfills the city's requirement is pertinent even though cost considerations could hamper its deployment.

An Intelligent Traffic Systems (ITS) is one of the critical transportation infrastructures that transportation agencies invest a huge amount of money to collect and analyze the traffic data to better utilize the roadway systems, improve the safety of transportation, and establish future transportation plans (WON, 2020).

Major objectives of ITS are to evaluate, develop, analyze and integrate new sensor, information and communication technologies, and algorithms to achieve traffic efficiency (Kumar, Rahman, Dhakad. 2021). The authors (Rida,Ouadoud, Hasbi & Chebli, 2018) states that knowing the traffic in real time is an essential element for the operation of the road network. To do this, we utilized a network of sensors where the role of each sensor is to collect a set of data in its environment and transmit it to a base station, which is responsible for traffic management (Rida, Ouadoud, Hasbi & Chebli, 2018).

As the number of vehicles has increased significantly, the capacity of existing transportation networks is almost at its maximum, causing severe traffic congestion in many countries (WON 2020). A traffic monitoring system is an effective alternative to mitigate traffic congestion. It is an integral component of Intelligent Transportation Systems (ITS) that is used to collect traffic data such as the number of vehicles, types of vehicles, and vehicle speed. Based on the collected data, it performs traffic analysis to better utilize the roadway systems, predict future transportation needs, and improve the safety of transportation (WON 2020).

Intelligent traffic system (ITS) applications provide a more efficient/low latency, effective, reliable, and safe driving experience that minimizes congestion with better traffic flow management. (Ekedebe, 2013). However, to achieve this, all ITS applications such as dynamic time allocation technology (DTAT), inter-vehicle communications (IVC), vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I) communications, situational awareness, Intelligent traffic light system (ITLS), and hard-breaking signals, and others must work synergistically or cooperatively to achieve optimization and efficient movement of traffic on streets and highways. Other technologies for making contemporary Intelligent Traffic Light System (ITLS) include deep reinforcement learning (Liang et al., 2019), Object Detection Systems (Sin-Chun & Chok-Pang, 2020) and different types of customized algorithms (Kawade, Deshmukh, Gamare & Sankhe, 2018).

Basically, DTAT leverages fuzzy logic and artificial neural network (ANN) to dynamically allocate time to each lane at a road junction. Fuzzy logic is an approach to variable processing that allows for multiple possible truth values to be processed through the same variable. It attempts to solve problems with an open, imprecise spectrum of data and heuristics that makes it possible to obtain an array of accurate conclusions. Fuzzy logic is designed to solve problems by considering all available information and making the best possible decision given the input (Scott, 2022).

Fuzzy logic stems from the mathematical study of multivalued logic. Whereas ordinary logic deals with statements of absolute truth (such as, "Is this object green?"), fuzzy logic addresses sets with subjective or relative definitions, such as "tall," "large," or "beautiful." (Anderson, 2022) This attempts to mimic the way humans analyze problems and make decisions, in a way that relies on vague or imprecise values rather than absolute truth or falsehood. In practice, these constructs all allow for partial values of the "true" condition. Instead of requiring all statements to be absolutely true or absolutely false, as in classical logic, the truth values in fuzzy logic can be any value between zero and one. This creates an opportunity for algorithms to make decisions based on ranges of data as opposed to one discrete data point. Today, fuzzy logic is used in a broad range of applications including: aerospace engineering, automotive traffic control, business decision-making, industrial processes, artificial intelligence, and machine learning.

Neural networks, also known as artificial neural networks (ANNs) or simulated neural networks (SNNs), are a subset of machine learning and are at the heart of deep learning algorithms. Their name and structure are inspired by the human brain, mimicking the way that biological neurons signal to one another (Krogh, 2021)

Inspired by biological neural networks, ANNs are massively parallel computing systems consisting of an extremely large number of simple processors with many interconnections. ANN models attempt to use some “organizational” principles believed to be used in the human. One type of network sees the nodes as ‘artificial neurons’. An artificial neuron is a computational model inspired in the natural neurons. Since the function of ANNs is to process information, they are used mainly in fields related with it. (Gupta, 2018) There are a wide variety of ANNs that are used to model real neural networks, and study behavior and control in animals and machines, but also there are ANNs which are used for engineering purposes, such as pattern recognition, forecasting, and data compression.

1.2 Problem Statement

Traffic congestion is an increasing problem in cities, urban and sub urban centers as people spend more of their time commuting. According to Sin-Chun and Chok-Pang, (2020) traditional traffic light control system causes a lot of problems such as long delays and congestion. The existing traffic signal system is in such a way that the signal is given for a lane using a timer circuit irrespective of the density of traffic in that lane (Jahnavi et al. 2021). Consequently, traffic situations of roads are not monitored dynamically in real-time, and the waiting period of a lane with a greater number of vehicles is very high while the waiting period of a lane with lesser number of vehicles is low. Furthermore, existing systems are ineffective in detecting traffic offenders and fining them (Liang et al. 2019). Thus, road traffic rules and regulations are not effectively enforced and vehicle drivers tend to violate traffic rules, which usually lead to many undesirable consequences including loss of productivity, fuel, time, fatigue and other health-threatening conditions. Furthermore, this creates opportunities for street urchins to extort and molest road users who are ignorant of their criminal activities.

1.3 Objectives of the Study

The main objective of this research work is to develop an intelligent traffic management system using Dynamic Time Allocation Technology (DTAT). This will be achieved by using the following specific objectives:

i. To model a road junction traffic scenario utilizing Unified Model Language (UML) tools.

ii. Design of a sensor network for the detection of vehicular presence and movement using motion sensors and IP cameras.
iii. Development of DTAT algorithms leveraging Fuzzy Logic and Artificial Neural Network (ANN) for efficient traffic management.
iv. Development of camera system for capturing vehicle plate number of offending drivers, as well as traffic offence SMS gateway for communicating with offenders and appropriate authorities.
v. System simulation and testing.

1.4 Justification of the Study

Intelligent road traffic system is of utmost importance to the entire society especially to those who live in major cities where traffic congestion is on the increase. This will facilitate the wellbeing of the people as average waiting time of vehicles on traffic queues will reduce, and productivity of the society improved as people shall spend less time on road traffics. Secondly, road traffic rules and regulations will be effectively enforced, and order maintained to a reasonable extent. Furthermore, fuel consumption of vehicles on the road will reduce, and the health condition and environmental status of the entire society will be highly improved as lesser Carbon II Oxide will be emitted, and people will suffer less fatigue due to long traffic queues.

1.5 Scope of the Study

This research work is focused on adopting software engineering principles in modeling and developing a real-time intelligent system for monitoring and controlling road traffic. The system utilized fuzzy logic and ANN to implement Dynamic Time Allocation Technology (DTAT). Also, circuits shall be designed to capture the vehicular traffic volume on the road using proximity sensors and microcontrollers, and also a camera system shall be simulated to capture and extract the license plate number of erring driver. An SMS gateway shall be developed to contact appropriate authorities as well as erring drivers for the purpose of fining them and charging fees for traffic offence.

Nevertheless, areas like circuit and solid state electronics such as semiconductors, transistors technologies, etc. are not covered. Neither the security of the embedded devices nor the communication network shall be considered in this project. Image analysis of captured road conditions is also not within the scope of this research.

CHAPTER TWO

LITTERATURE REVIEW

2.1 Review of Key Concepts on Intelligent Traffic Systems (ITS)

The traffic light has integrated technologies such as fuzzy logic Artificial Neural Network (ANN), Internet of Things (IoT) to improve on time allocation to different lanes (Vogel, Oremovic & Simic, 2018). Thus, key concepts on Intelligent Traffic Systems (ITS) are thoroughly reviewed in this section. The review is grouped in four sections, namely: Intelligent Traffic-Based Review, IoT-Based Review, Fuzzy Logic-Based Review and Artificial Neural Network-Based Review.

2.1.1 Intelligent Traffic-Based Review

An Intelligent Traffic Systems (ITS) is one of the critical transportation infrastructures that transportation agencies invest a huge amount of money to collect and analyze the traffic data to better utilize the roadway systems, improve the safety of transportation, and establish future transportation plans (WON, 2020).

The goals of ITS are to evaluate, develop, analyze and integrate new sensor, information and communication technologies, and algorithms to achieve traffic efficiency (Kumar, Rahman & Dhakad, 2021). Rida, Ouadoud, Hasbi and Chebli (2018) stated that understanding the traffic in real time is an important factor for the operation of the road network. To do this, a network of sensors was utilized, and the role of each sensor is to gather a set of data in its area and send it to a base station, which is responsible for managing traffic (Rida et al, 2018).

When the number of vehicles has risen very much, the function of the existing transportation networks is almost at its peak, resulting to severe traffic congestion in many places (WON 2020). In order to mitigate traffic congestion, a traffic monitoring system is needed. It is an essential element of Intelligent Transportation Systems (ITS) that is used to gather traffic data such as types of vehicles, the number of vehicles, and speed of each vehicle. Based on the data gathered, it carries out traffic analysis to better utilize the roadway systems, forecast the needs of future transportation, and enhance transport safety (WON 2020).

A review of the state-of-the-art on traffic monitoring systems focusing on the major functionality-vehicle classification was presented by Won (2020). He organized various vehicle classification systems, examined research issues and technical challenges, and discussed hardware/software design, deployment experience, and system performance of vehicle classification systems. Finally, he discussed a number of critical open problems and future research directions in an aim to provide valuable resources to academia, industry, and government agencies for selecting appropriate technologies for traffic monitoring applications.

2.1.2 Internet of Things (IoT) Based Review

The traffic light has integrated Internet of Things (IoT) technologies such as wireless sensor network, cameras, etc. in a bid to know the volume of traffic on each lane of the junction (Rida et al, 2018). Internet of things (IoT) which is used to get machines communicating with machines has also been deployed on the traffic light in a bid to improve the traffic light operation (George, 2018; Mittal & Chawla, 2020). The concepts of some of the technologies utilized in IoT are discussed in the following paragraphs.

2.1.2.1 Vehicular Ad Hoc Network (VANET)

The traffic light is used to control vehicular movement and several traffic lights can be on a major road having different intersections. This leads to the traffic lights and the vehicles to be seen as a kind of network referred to as vehicle ad hoc network (VANET). The traffic light is an integral part of VANET.

VANET is a technology that accepts intelligent vehicle as a node and uses these nodes to build a mobile network (Adak & Balta, 2021). It (VANET) is an emerging technology that aims to provide wireless communication between moving vehicles, as well as between vehicles and infrastructure stations (Goudarzi & Asgari, 2018). In vehicular networks, communication between vehicles is provided via OBUs (onboard units) embedded in the vehicle. OBUs have 802.11p antennas that use the DSRC (Dedicated Short Range Communications) standard created by modifying the frequency and power parameters of 802.11a. The DSRC standard has a bandwidth of 75 MHz in the 5.9 GHz band and consists of 7 channels each of 10 Mhz. Average data rates on channels are between 6-27 Mbps. The vehicles broadcast the obtained data, such as position, speed, direction, neighborhood table, etc., to other vehicles (V2V) or roadside units (RSU-V2I) via beacon signals (Adak & Balta, 2021).

(Goudarzi & Asgari, 2018) notes that VANETs have some characteristics that differentiate them from other types of mobile ad hoc networks (MANETs). These characteristics include fast node movements, a large network, and constrained mobility imposed by the road topology. Owing to such differences, topology-based MANET routing protocols, such as AODV, OLSR, and DSR, perform less efficiently in VANETs (Goudarzi & Asgari, 2018). But the movement of vehicles causes the communication links between vehicles to be broken frequently. This is a major setback in VANET.

2.1.2.2 Sensors to Detect the Volume of Traffic on the Road

For the traffic light to be more effective, it has to factor in the volume of traffic per lane as part of its decision-making process. There are different types of sensors that can be deployed for this purpose. (WON 2020) observes that some of these sensors could be mounted on the road, some by the side of the road while others are mounted over the road. These sensors detect the presence of vehicles on the road and more vehicles are tantamount to increased traffic volume.

A. Loop Detectors

An inductive loop detector is one of the most commonly used traffic monitoring systems for vehicle detection. It is a coil of wire that is embedded under the road surface. It captures the change of inductance and generates a time-variable signal when a vehicle passes over it. The characteristics of the signal such as the amplitude, phase, and frequency spectrum are varied depending on the type of vehicle. These unique characteristics of the signal are known as the magnetic profile and can be used to classify the vehicle that has just been detected (WON 2020).

B. Magnetic Sensors

The large amount of ferrous metal in a vehicle frame induces disturbance to the earth's magnetic field. Magnetic sensors are used to detect vehicles by capturing the distinctive changes in the magnetic field which depend on different vehicle body types (WON 2020).

C. Vibration Sensors

Using highway pavement itself as a transducer, vibration sensors capture the unique vibration patterns induced by passing vehicles due to the low elasticity of road pavement that makes vibrations well localized in time and space. By this method, vehicles on the road are detected and even the type of vehicle can be known.

D. Acoustic Sensors

The acoustic sensor-based vehicle detection systems capture the audio signal induced by a passing vehicle using microphone sensors. The success of these types of solutions depends largely on effective feature extraction from acoustic signals. However, since the performance of the acoustic sensor is easily affected by ambient noise, it is very challenging to identify vehicle presence and also to identify the type of vehicle.

E. Light Detection and Ranging Sensor

A light detection and ranging (LIDAR) sensor sends eye-safe laser lights and record the receptions to calculate the points of the environment such as the road, passing vehicles, and vegetation, *etc.* Based on the collected data, features are extracted such as the size and shape of a passing car. LIDAR is especially powerful in identifying the shape of a passing car due to its high precision sensing. However, the vehicle occlusion problem remains as a critical challenge for LIDAR-based vehicle detection systems.

F. RADAR

The basic mechanism of radar-based vehicle detection systems is that the radar-based systems exploit the receptions of radio signals from the vehicle body. The radar sensors use radio waves. They are less vulnerable to weather and light conditions than LIDAR, but the latter provide more accurate representation of the vehicle body.

G. Wi-Fi Transceivers

Recently, Wi-Fi-based traffic monitoring systems have been developed specifically targeting the endemic cost issue for deploying a large number of traffic monitoring systems to cover huge miles of rural highways. The idea is to leverage the unique Wi-Fi channel state information (CSI) patterns induced by passing vehicles to perform detection and identification (WON, 2020).

H. Cameras

The most widely adopted sensor for vehicle detection systems is the camera (WON, 2020). A camera provides rich information for identification such as the visual features and geometry of passing vehicles. A single camera is sufficient for detecting vehicles in multiple lanes. Advanced image processing technologies supported by sufficient processing power allows for identification of multiple vehicles very quickly and accurately.

I. Ultrasonic Sensors

Vehicle detectors that use ultrasonic sensor deploy the concept of echolocation to measure the distance between the sensor and the road surface. To detect the presence of a vehicle, the measured distance is altered by the vehicle which is positioned between the road and the sensor. (Putra & Warnars, 2018) used ultrasound sensor for vehicle detection in a traffic management scenario.

J. Passive Infra-Red (PIR) Sensors

When the engine of a vehicle is on, that vehicle gives off heat. This heat can be detected by PIR sensor. Vehicle detectors using this technology simply detect the heat coming off vehicles as they move on the road. (Putra, Warnars, 2018) used this sensor for vehicle detection for traffic management.

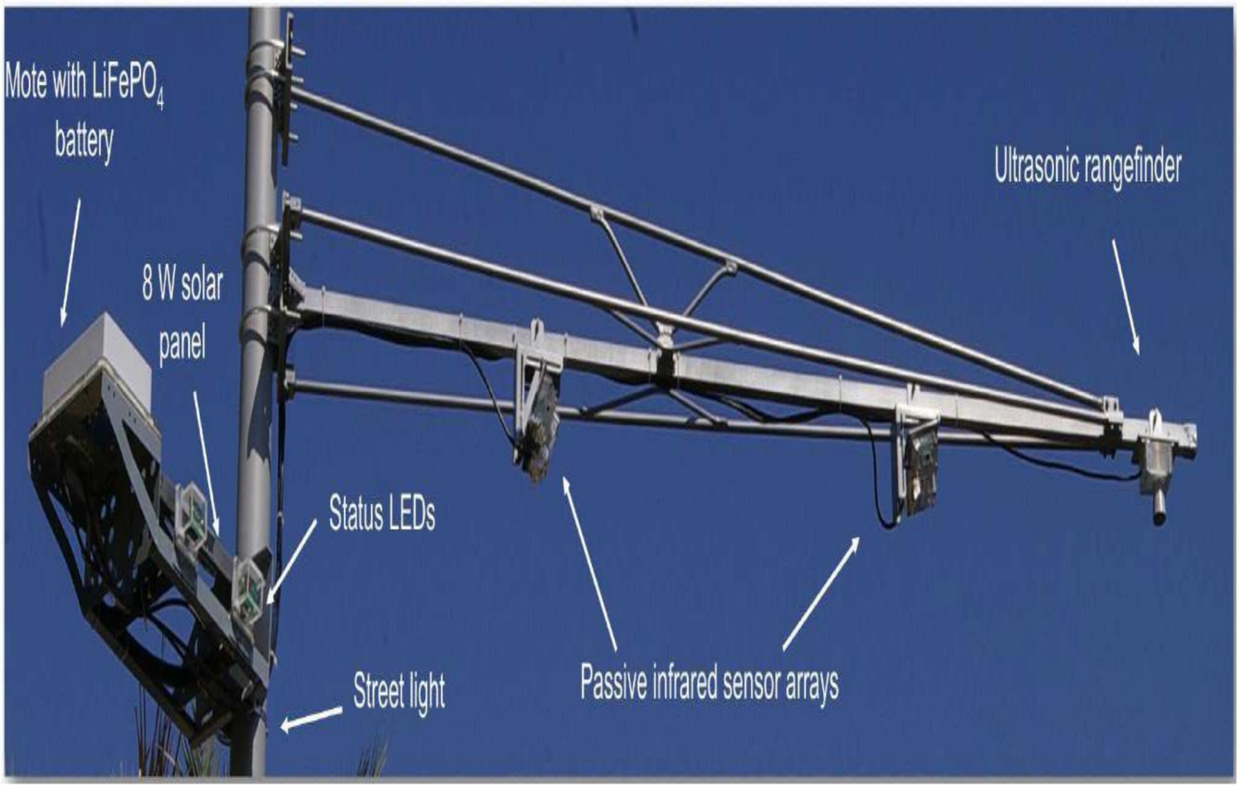

Plate 2.1: Vehicle detection system using ultrasound and PIR sensors (WON, 2020).

The figure 2.1 represents taxonomy of sensors used for traffic monitoring considering where they are installed; in the road, over the road or by the side of the road.

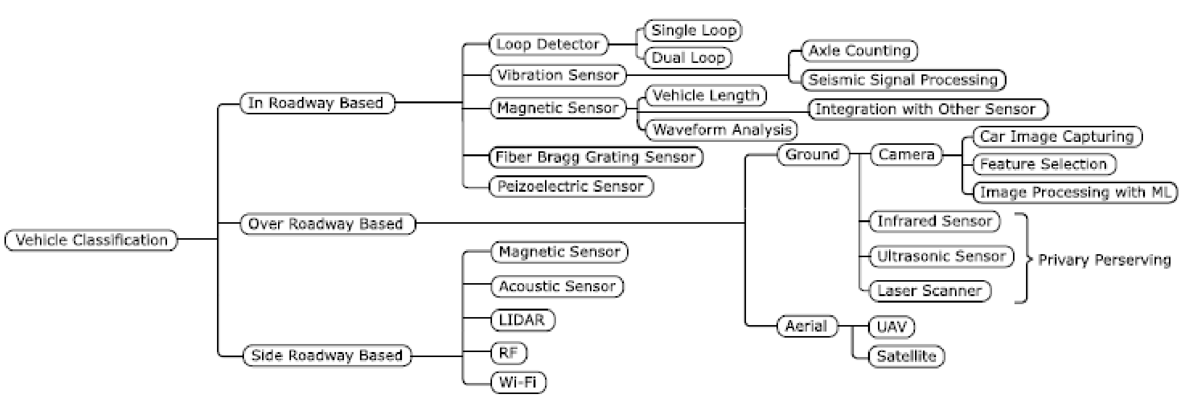

Fig. 2.1: Taxonomy of sensor technologies for vehicle detection (WON, 2020).

2.1.2.3 License Plate Identification Technology

The license plate of a vehicle is a unique identifier for the vehicle (Nguyen, Kieu, Wen & Cai, 2018); with it, the vehicle can be traced anywhere. In Nigeria, the Federal Road Safety Corpse (FRSC) keeps a database of all the vehicle license plates in the country as every vehicle is registered with them. They are equally responsible for the renewal of vehicular license. The FRSC has a record of the contact particulars – phone number, house number etc., of the owner of the registered vehicle. Hence, the FRSC database is pertinent in identifying the owner of any vehicle and communicating with him.

Once the camera at the traffic light snaps the license plate of the offending vehicle, several technological advancements can be applied to the image with which the cameral can uniquely single out the license plate number of the vehicle. (Sharma, Kumar, Tiwari, Yadav & Nezhurina, 2018) notes that in automatic license plate recognition systems 4 steps are involved - The first step is to capture the image of a vehicle using a camera. The second stage is to separate out the license plate region which is an area of interest, in the captured image. The third step is to segment the characters from the extracted area of interest. Lastly, the segmented characters are recognized on the basis of the best match of their features with the help of a classifier. Nguyen, Kieu, Wen and Cai, (2018) identified the different areas of LPR (License Plate Recognition) system to include auto toll collection, traffic monitoring, auto gate systems and many other intelligent and transportation systems.

Sharma et al. (2021), the authors applied convolutional neural network and support vector machine-based concept to license plate recognition. Using the camera image of the vehicle, the characters are extracted and segmented. Then features of the segmented characters are extracted. The extracted features are classified using convolutional neural networks and support vector machine for the final recognition of the license plate. Using this technique, the authors were able to achieve 96.5% recognition rate using the above method.

Rehman, Rathore, Paul, Saeed and Ahmad, (2018) considered that License plate recognition (LPR) systems are used to automatically extract the characters from license plates positioned in front of a camera. Therefore, the geometric detection pattern is the region within which the LPR system can accurately recognize license plates with minimal errors. The authors note that multiple factors contribute to the shape and size of the geometric detection pattern.

Amin and Payman (2019) observed that for most License plate recognition (LPR) systems, the resolution of input images is not very high because of high cost of high-resolution cameras. This can lead to LPR algorithm not working well. To improve on the situation, super-resolution (SR) techniques are widely used to construct a high-resolution (HR) image from several observed low-resolution (LR) images, thereby improving on the image quality caused by the use of low-resolution camera. Most low-resolution cameras snap successive frames from a target; hence the authors used a SR technique based on POCS (Projection onto Convex Sets) to reconstruct a HR license plate image from a set of LR images. The authors used a two-step image matching algorithm to improve the quality of image registration stage. For the first step, they used the Fourier-Mellin image matching for registration which overcomes the scale and rotation challenge. For the second step, they used the Keren or Vandewalle image matching to improve the quality of final image.

2.1.2.4 Short Message Service (SMS) Gateway Technology

The use of the GSM phone has traversed diverse fields of endeavor and its ubiquitous applications are possible primarily because of size – it is always in the hands or pocket of every adult. It is at the center of mobile communications technology. It is the means through which almost everybody can be contacted. According to Nigerian Communication Commission NCC, there are over 100 million GSM phones in Nigeria as at 2020.

Many devices and applications have been developed to send SMS or communicated with SMS and these devices have a GSM module with which they access the SMS gateway that enables them to send SMS to preprogrammed phone numbers. (Alhamri & Dianta, 2019) designed an automatic drain system for freshwater fish pond and used an SMS gateway to effect monitoring of the system. (Kamelia et al., 2018) implemented an electronic key (E-key) system using SMS technology. (Husni & Hidayat, 2018) used SMS gateway to implement an e-payment system. (Ahmadian et al., 2018) used it as the core information backbone in a system that checks on the progress of a construction project. Hence, SMS gateway can be applied to an Intelligent Traffic Management System as a means to contact traffic offenders and fine them accordingly.

2.1.2.5 Traffic Offence Mitigation

UKEssays (2018) notes that as drivers wait long on the queue in the traffic light, they might get irritated. This behavioral condition might prompt the driver in the vehicle in front of the traffic light to beat the red light. This is a traffic offence and something must be done to mitigate it. This type of traffic offence is very pronounced in Nigerian road. In other to mitigate this type of traffic offence, the license plate number identification technology comes to play. This involves the use of camera to take several snapshots of the offending vehicle. The images so acquired by the camera are passed through a license plate identification algorithm in other for the intelligent traffic management system to retrieve the license plate number of the vehicle. Also, the intelligent traffic management system will extract the driver's phone number from the appropriate database and this phone number is used to contact the errant driver and also to communicate fines to him.

2.1.3 Fuzzy Logic-based Review

Fuzzy logic is an approach to variable processing that allows for multiple possible truth values to be processed through the same variable. It attempts to solve problems with an open, imprecise spectrum of data and heuristics that makes it possible to obtain an array of accurate conclusions. Fuzzy logic is designed to solve problems by considering all available information and making the best possible decision given the input (Scott, 2022). Fuzzy logic stems from the mathematical study of multivalued logic. Whereas ordinary logic deals with statements of absolute truth (such as, "Is this object green?"), fuzzy logic addresses sets with subjective or relative definitions, such as "tall," "large," or "beautiful" (Anderson, 2022). This attempts to mimic the way humans analyze problems and make decisions, in a way that relies on vague or imprecise values rather than absolute truth or falsehood. In practice, these constructs all allow for partial values of the "true" condition. Instead of requiring all statements to be absolutely true or absolutely false, as in classical logic, the truth values in fuzzy logic can be any value between zero and one. This creates an opportunity for algorithms to make decisions based on ranges of data as opposed to one discrete data point. Today, fuzzy logic is used in a broad range of applications including: aerospace engineering, automotive traffic control, business decision-making, industrial processes, artificial intelligence, and machine learning. This is illustrated in Figures 2.2 and 2.3.

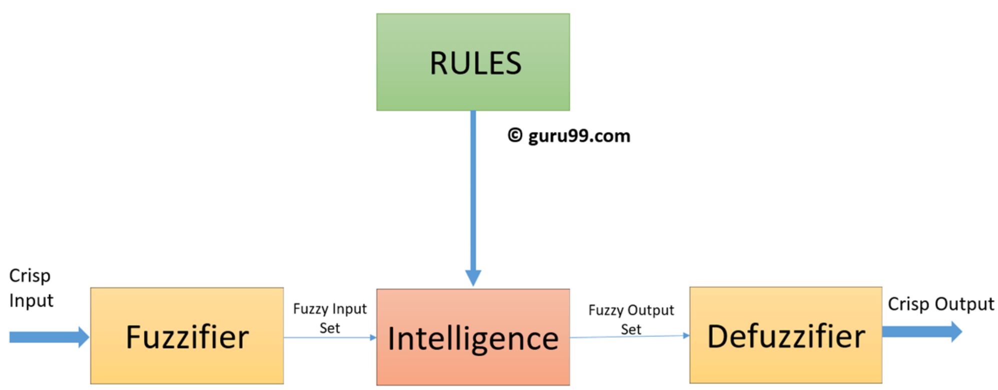

Figure 2.2: Fuzzy logic System

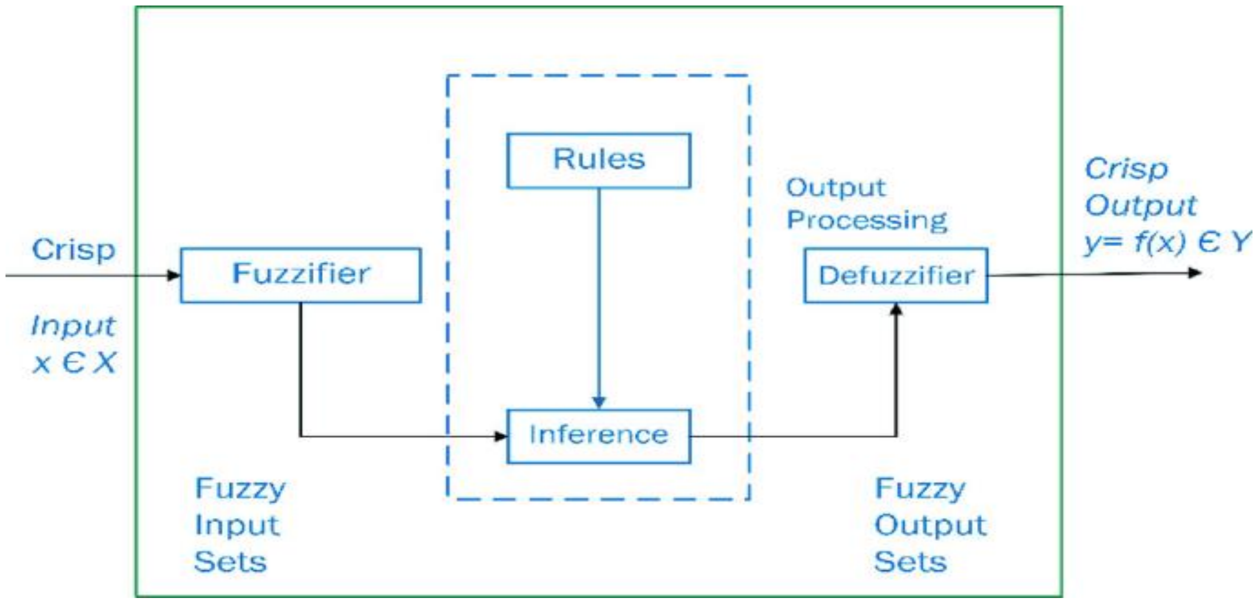

Figure 2.3: Architecture of Fuzzy Logic System

2.1.4 Artificial Neural Network Based Review

Neural networks, also known as artificial neural networks (ANNs) or simulated neural networks (SNNs), are a subset of machine learning and are at the heart of deep learning algorithms. Their name and structure are inspired by the human brain, mimicking the way that biological neurons signal to one another (Krogh, 2021)

Inspired by biological neural networks, ANNs are massively parallel computing systems consisting of an extremely large number of simple processors with many interconnections. ANN models attempt to use some "organizational" principles believed to be used in the human. One type of network sees the nodes as 'artificial

neurons'. These are called artificial neural networks (ANNs). An artificial neuron is a computational model inspired in the natural neurons. Since the function of ANNs is to process information, they are used mainly in fields related with it. (Gupta, 2018) There are a wide variety of ANNs that are used to model real neural networks, and study behavior and control in animals and machines, but also there are ANNs which are used for engineering purposes, such as pattern recognition, forecasting, and data compression. These basically consist of inputs (like synapses), which are multiplied by weights. Weights assigned with each arrow represent information flow. These weights are then computed by a mathematical function which determines the activation of the neuron. Another function (which may be the identity) computes the output of the artificial neuron (sometimes in dependence of a certain threshold). The neurons of this network just sum their inputs. Since the input neurons have only one input, their output will be the input they received multiplied by a weight. This is illustrated in Figure 2.4.

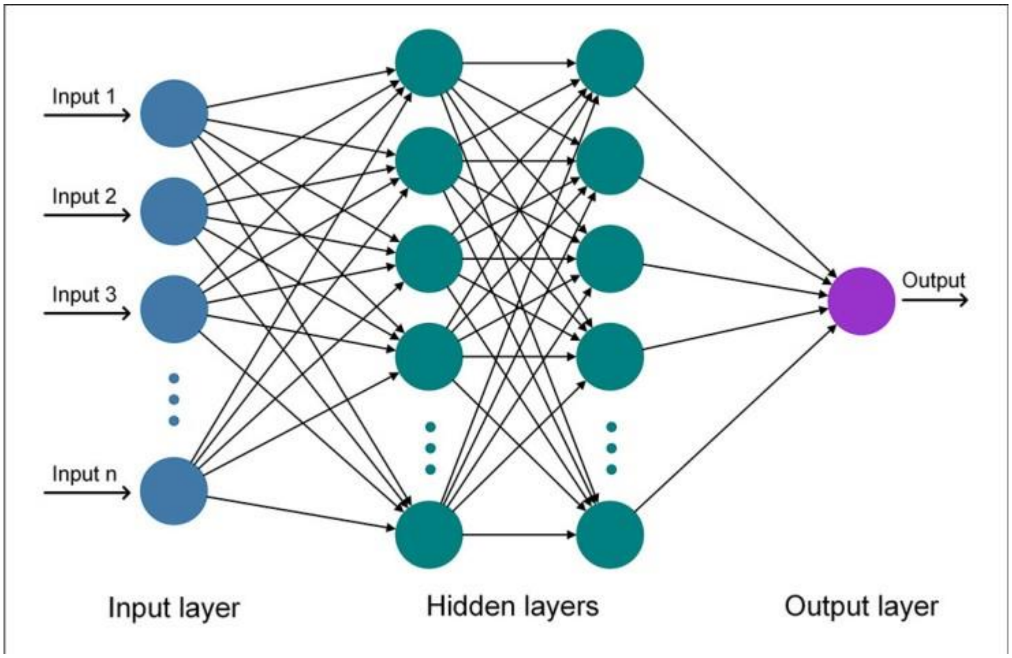

Figure 2.4: Schematic diagram of an Artificial Neural Network (ANN)

2.2 Review of Related Works

Related works on Intelligent Traffic Systems (ITS) and Traffic Monitoring Systems (TMS) are reviewed in this section. The review is grouped in four sections, namely: Intelligent Traffic-Based Review, IoT-Based Review, Fuzzy Logic-Based Review and Artificial Neural Network-Based Review.

2.2.1 Related Works on Intelligent Traffic Systems

Latif et al. (2018) proposed an intelligent traffic monitoring system using graph theory and formal methods. The proposed model has various nodes that are assumed within a city including roads, objects and traffic signals to make a collective intelligent traffic guidance and monitoring system. Operations performed by the systems includes finding shortest path in terms of time and distance, finding specific area within city and finding safest and low rush ways towards the destination. The graph-based model was easily transformed into a formal model using Vienna Development Method-Specification Language (VDM-SL). The proof of correctness was provided by various facilities available in the VDM-SL toolbox.

A common protocol by which there could be vehicle-to-vehicle and vehicle-to-road communications ideal for avoiding collisions and road accidents, all in a vehicular ad hoc network (VANET), was designed by Zadobrischi and Dimian (2021). Ways of transmitting warning messages simultaneously by vehicle-to-vehicle and vehicle-to-infrastructure communications by various multi-hop routings were set out. Approaches on how to improve communication reliability by achieving low latency were addressed through the multi-channel (MC) technique based on two non-overlaps for vehicle-to-vehicle (V2V) and vehicle-to-road (V2R) or road-to-vehicle (R2V) communications. The contributions of the work offered an opportunity to use common communication adaptable protocols, depending on the context of the situation, coding techniques, scenarios, analysis of transfer rates, and reception of messages according to the type of protocol used. Communications between the road infrastructure and users through a relative communication protocol were highlighted and simulated. The results obtained by the proposed and simulated scenarios demonstrate that it is complementary and that the common node of V2V/V2R (R2V) communication protocols substantially improves the process of transmitting messages in low-latency conditions and is ideal for the development of road safety systems.

Lau et al. (2022) proposed a traffic monitoring system which is PC-based, no infrastructure requirement but suitable for the complex road networks in Hong Kong. Besides, the system was designed to recognize vehicle type and plate number for vehicles commonly used in Hong Kong. By analyzing the data collected on road usage, the authors presented their findings for decision-making regarding traffic policies.

A distributed cooperative backpressure-based traffic light control method was proposed in (Hao, Yang, Ding, & Guo, 2019). The urban traffic network was modeled as a smart agent-controlled queuing network, in which the intersection agents exchange the queue length information and the selected activating light phase information of neighboring intersections through communications and determine the activating light phase at each time slot according to local traffic information. Light phase switching coordination among adjacent intersections was achieved using the consensus-based bundle algorithm, in which the cooperative light phase switching problem is viewed as a task assignment issue among adjacent intersections. Simulation results illustrated that the proposed cooperative backpressure-based traffic light control method obtained better performance than the original backpressure-based and fixed-time traffic control methods.

Pahal et al. (2019) focused on an ontology-based context-aware framework for providing services such as smart surveillance and intelligent traffic monitoring, which employ IoT technologies to ensure better quality of life in a smart city. An IoT network combines the working of Closed-circuit television (CCTV) cameras and various sensors to perform real-time computation for identifying threats, traffic conditions and other such situations with the help of valuable context information. Multimedia Web Ontology Language (MOWL) was employed for semantic interpretation and handling inherent uncertainties in multimedia observations linked with the system. MOWL also allows for a dynamic modeling of real-time situations by employing Dynamic Bayesian networks (DBN), which suits the requirements of an intelligent IoT system. In the proposed approach, continuous video stream of data captured by CCTV cameras can be processed on-the-fly to give real-time alerts to security agencies. These alerts can be disseminated via e-mail, text messaging, on-screen and alarms not only to pedestrians and drivers in the locality but also the nearest police station and hospital in order to prevent and decrease the loss incurred by any event.

Zheng et al. (2019) focused on the short-term traffic flow prediction problem based on real-world traffic data as one critical component of a smart city. They developed a novel ensemble model (EM) based on long short-term memory (LSTM), deep autoencoder (DAE) and convolutional neural network (CNN) models. Their approach took into account both temporal and spatial characteristics of the traffic conditions. They used two real traffic data (California and London roadways) with different characteristics to train and test the models. Their results indicate that the proposed ensemble model achieves the most accurate predictions (approx. 97.50% accuracy) and is robust against high variance traffic flow.

Bharani (2018) focused on reviewing the RFID and Sensor based technologies that are implemented in Smart Traffic System (STS) which concentrating only on traffic density management in the traffic scenarios and not meant for the emergency situations. Also, the work proposed an innovative approach on Traffic management with the help of the integration of IoT and smart phone applications including the Geo Fencing.

Tchappi et al. (2020) presented a critical review of the use of holonic paradigm in order to model and simulate traffic and transportation systems. They also surveyed existing works using the holonic paradigm for traffic and transportation applications. This was followed by a detailed analysis of the results of the survey. In particular, the relevance, the design approaches and the holonification orientation methodologies were investigated. Finally, based on the extensive review, open issues of holonic paradigm in modeling and simulation of traffic and transportation models were highlighted.

Lima (2019) focused on road transportation, and the ability of classifying a passing vehicle, which plays a key role in traffic monitoring applications and tolling systems. The proposed system relies on a simple photo sensor to classify a passing vehicle. The sensor captures and measures the light emitted by a vehicle’s headlights and then extracts its features and classify them as a signature of the perceived light variation. A neural network is trained to perform the classification process. The prosed method was implemented on a SoC and evaluated on a real-life scenario, including two most common types of vehicles: cars and buses. Preliminary results show that the system was able to correctly classify vehicles in real-time with an accuracy of nearly 80%.

A new approach for controlling Traffic System was designed by Nagaraju et al. (2019). The proposed system was designed with components like Raspberry Pi, Pi-Camera, RFID, IR sensors. Raspberry Pi is the main component which is used to control all, it acts like a controller. Density of the traffic will be decided with the help of IR sensors, IR Sensor checks the vehicles passing on that particular path. It also determines the network congestion, and hence changes the green light duration for that path.

To improve the overall network efficiency, Xu et al. (2019) developed an online agent-based signal coordination scheme, underpinned by the communication among different intersection control agents. In addition, the initial coordination scheme that pre-adjusts the offsets between the intersections was developed based on the historical demand information. Comparison and sensitivity analysis were conducted to evaluate the performance of the proposed method on a customized traffic simulation platform using MATLAB and VISSIM. Simulation results indicate that the proposed method can effectively avoid network oversaturation and thus reduces average travel delay and improves average vehicle speed, as compared to rule-based multi-agent signal control methods.

Lu et al. (2021) presented a monitoring scheme based on connections between human and machine (TTS-HCI) and Bayesian filters. They also discussed current intelligent transport networks and possible complications caused by human–computer interaction. The experimental analysis of context-aware of the HCI for Transport system in Urban Areas results was obtained as the traffic congestion ratio was 87.6%, increasing the roadways ratio was 87.12%, and the human safety ratio was 84.5%, speed control ratio was 82.25%, vehicle tracking ratio was 93.80%.

Abdoos and Bazzan (2021) employed a hierarchical multi-agent system including two levels to control traffic signals. Each traffic signal was controlled by an agent that sits in the physical level, i.e., in the first level. For the other levels, the traffic network was divided into a number of regions, each controlled by a region controller agent. The first level agents utilize reinforcement learning to find the best policy, while they send their local information to the above level agents. The local information was used to train a long short-term memory (LSTM) neural network for traffic status prediction. The agents in the above level can control the traffic signals by finding the best joint policy using the predicted traffic information. Experimental results proved the effectiveness of the proposed method in a traffic network including 16 intersections.

Latif et a. (2018) focused on intelligent traffic monitoring system using graph theory and formal methods. The proposed model has various nodes that were assumed within a city including roads, objects and traffic signals to make a collective intelligent traffic guidance and monitoring system. Many operations including finding shortest path in terms of time and distance, finding specific area within city and finding safest and low rush ways towards the destination were designed. The graph-based model was transformed into a formal model using Vienna Development Method-Specification Language (VDM-SL). The proof of correctness is provided by various facilities available in the VDM-SL toolbox.

Romero-Ania et al. (2021) proposed a consistent combination of the ELECTRE TRI multiple criteria decision sorting method and the DELPHI procedure, the objective of which was to identify which urban public transport vehicles are acceptable, taking into consideration a suggested sustainable threshold, which includes economic and environmental strict requirements. The proposed model was based on 2020 Madrid urban public road transport data, published by Madrid City Council, which were compiled by the authors, and assessed by a panel of 20 experts to identify criteria and factors included in the model. Findings help local administrations to identify which urban public transport vehicles should be progressively replaced by those classified as economically efficient and additionally environmentally sustainable.

Dong et al. (2021) proposed a new hybrid model based on reinforcement learning for the accurate forecasting of traffic speed. The model contains the LSTM network and the

GRU network as predictors for in-depth mining of the characteristics of traffic speed data and uses reinforcement learning to integrate the results of the two predictors, combining the advantages of multiple predictors to achieve stable and accurate forecasting results of traffic speed. The work used two sets of measured traffic data from Guangzhou to test the effectiveness, and five other traffic speed forecasting models were also established for comparison. Experimental results show that the hybrid model applied in the work was the best performance on both data sets, and the MAPEs are 5.02% and 3.25%.

Lin et al. (2020) proposed a prediction model (called pattern sensitive network) that can handle different traffic patterns automatically. By using adversarial training, the proposed model can make more accurate predictions in unusual states without compromising its performance in usual states. Experiments demonstrate that the model can work well in both usual traffic states and unusual traffic states.

A method for traffic pattern analysis and state prediction for correlated routes in the road network was proposed by Z. Zhang and W. Zhang (2020). First, the concepts of correlated route, correlated route chains, and correlated route sets were defined, and a route correlation degree calculation model that considers route traffic heterogeneity and its judgment criteria was proposed to determine the correlated route sets in the region. Second, a self-organizing mapping (SOM) algorithm with Dunn index (DI), named as SOM_DI, was incorporated to classify the traffic states on the correlated route chain and determine the optimal number of traffic state. The traffic pattern on the correlated path chain was analyzed to obtain the temporal state chains and the spatial state chains. Finally, an algorithm was proposed to select the input spatio-temporal features of the support vector regression (SVR) model and predict the traffic state on the correlated route chain, which was named as STFS_SVR. The simulation results show that the method proposed can accurately classify the correlated routes of regional traffic and its optimal traffic state.

Wang et al. (2018) provided an overview of several promising research areas for traffic management in Social Internet of Vehicles (SIoV). Given the significance of traffic management in urban areas, they investigated a crowdsensing-based framework to provide timely response for traffic management in heterogeneous SIoV. The participant vehicles based on D2D communications integrated trajectory and topology information to dynamically regulate their social behaviors according to network conditions. A real-world taxi trajectory analysis-based performance evaluation was provided to demonstrate the effectiveness of the designed framework.

A Dynamic and Intelligent Traffic Light Control System (DITLCS) which takes real-time traffic information as the input and dynamically adjusts the traffic light duration was proposed by Kuma et al. (2020). The proposed DITLCS runs in three modes namely Fair Mode (FM), Priority Mode (PM) and Emergency Mode (EM) where all the vehicles are considered with equal priority, vehicles of different categories are given different level of priority and emergency vehicles are given at most priority respectively. Furthermore, a deep reinforcement learning model was also proposed to switch the traffic lights in different phases (Red, Green and Yellow), and fuzzy inference system selects one mode among three modes i.e., FM, PM and EM according to the traffic information. The authors evaluated DITLCS via realistic simulation on Gwalior city map of India using an open-source simulator i.e., Simulation of Urban MObility (SUMO). The simulation results proved the efficiency of DITLCS in comparison to other state of the art algorithms on various performance parameters.

Rida et al. (2018) proposed an adaptive traffic light control method for an isolated intersection that considers a number of traffic factors such as traffic volume, and the waiting time. In applying this method priority was given to the shortest queue. Experimental results demonstrate that the proposed algorithm reduces the waiting time of vehicles compared to a fixed time control algorithm.

A novel TLCS, named PALM, for tackling the challenge for handling mixed traffic scenarios was proposed in (Tan et al., 2021). PALM considers the traffic flow at each intersection and adjacent ones and adjusts the traffic lights schedule for the next few phases accordingly. It also optimizes the signal timing and phases to better serve the platoons formed by CAV. The simulation results show that the approach achieved up to

75.34% and 33.02% drop in the average waiting time compared to the static and actuated TLCS, respectively.

Grag et al. (2018) addressed the problem of congestion around the road intersections. They developed a traffic simulator to optimally simulate various traffic scenarios, closely related to real-world traffic situations. They contended that adaptive real-time traffic optimization is the key to improving existing infrastructure's effectiveness by enabling the traffic control system to learn, adapt and evolve according to the environment it is exposed to. They also put forward a vision-based, deep reinforcement learning approach based on a policy gradient algorithm to configure traffic light control policies. The algorithm was fed real-time traffic information and aimed to optimize the flows of vehicles travelling through road intersections. Preliminary test results demonstrate that, as compared to the traffic light control methodologies based on previously proposed models, configuration of traffic light policies through this novel method is extremely beneficial.

A framework of applying Vehicle-to-vehicle communications (V2V), Vehicle-to-infrastructure (V2I), Vehicle-to-everything (V2X), Internet of things (IoT) and Artificial intelligent techniques (AI) in the context of traffic lights management and control in an Internet of Things environment was developed by Abohashima et al. (2020). The dynamic scheduling of traffic lights given the real-time data from road and vehicle embedded sensors was also elaborated. Furthermore, the authors integrated the mathematical methods with the Neuro-Fuzzy based traffic control system for taking an intelligent decision based on the present traffic flows.

The research by Astarita et al. (2018) explored the use of floating car data to synchronize traffic signal systems. They suggested that new technologies are going to change the future of traffic signal control and management and possibly will introduce new traffic signal system that will be based on floating car data (FCD). This will turn into an increased sustainability of transportation in terms of energy efficiency, traffic safety and environmental issues. Thus, they thoroughly studied the concept of floating car data (FCD) in systems that had the objective of gaining traffic information and/or signal information. Furthermore, traffic signal synchronization based on FCD was extensively studied in terms of cooperative-competitive paradigm between “instrumented” vehicles and conventional vehicles. Finally, they provided results of a first research into the possibilities of using FCD for traffic signal synchronization and the consequent cooperation-competition between different vehicles in the system.

Gusrialdi et al (2018) proposed a resilient cooperative control design for networked cooperative systems when subjected to external attacks. The systems considered in the work can have any information topology described by a leader-follower digraph. A potential attack on such systems consists of unknown bounded signals generated from any linear or nonlinear finite-L 2 -gain exogenous dynamical system and distributed into nodes of the system's network. The purpose of the attack is to destabilize the consensus dynamics by intercepting the system's communication network and corrupting its local state feedback. The proposed resilient control design introduced a virtual system with hidden network such that the overall system consisting of the original consensus system, the virtual system, and the attack dynamics is stable without requiring any information about the locations or nature of the attack. This was accomplished by utilizing the concept of competitive interaction to provide explicit design criteria for the hidden network of the virtual system to interact with the original system. Graph theoretical approach and Lyapunov direct method were used to analyze the overall system, and the result obtained show that the proposed design ensures stability of the overall system and preserves the consensus of the original system.

In order to improve vehicular traffic flow in dense urban areas, (Guidoni et al., 2020) presented a new Vehicular Traffic Management Service based on Traffic Engineering theory, called Re-RouTE. The Re-RouTE service relies on the density of vehicles in roads and applies the flow-density macroscopic traffic engineering model to identify congested routes. Moreover, the service was designed to reduce traffic jams instead of moving them to a different road/area. Simulation results show Re-RouTE improved travel time, travel distance, speed and the number of messages transmitted when compared to a literature solution.

Rehman et al. (2018) proposed a framework to enhance the efficiency of the ant colony optimization (ACO) algorithm to optimize vehicular traffic, i.e., named as smart traffic distribution ACO. It helped to optimize the route and city traffic efficiently while avoiding congestion in all circumstances using up-to-date city traffic data. The proposed framework finds the optimal path in such a way that the traffic flow on each road remains normal. The detection of congestion on the road at an early stage and even distribution of traffic on all roads helps to achieve maximum flow, speed, and optimum density of the roads.

2.2.2 Related Works on Internet of Things

Dubey et al. (2017) proposed an adaptive traffic system which is connected to internet so that different lanes can be monitored constantly. The data obtained from different lanes were examined and controlled by Central Traffic Control Office from one place. Data obtained thus gives value of traffic congestion in particular lane, according to which traffic lights are programmed to work. If the first lane is having less traffic than other lane, then the signal lights will be decided on the basis of less wait time and less pollution. The system is useful in emergency and VIP clearance and in traffic survey. This increases the efficiency of traffic clearance. This also reduces pollution and traffic congestion, thus being an Adaptive Traffic Control System using Internet of Things.

Jaiswal and Anand (2020) designed an energy-efficient routing protocol for wireless sensor network based IoT application having unfairness in the network with high traffic load. The proposed protocol considers three-factor to select the optimal path, i.e., lifetime, reliability, and the traffic intensity at the next-hop node. Rigorous simulation was performed using NS-2. Also, the performance of the proposed protocol was compared with other contemporary protocols. The results show that the proposed protocol performs better concerning energy saving, packet delivery ratio, end-to-end delay, and network lifetime compared to other protocols.

A novel traffic management system using IoT and augmented reality was presented by Dey et al (2019). The intelligent vehicles were embedded with sensor infrastructure for sensing environment. The system was applied for safe navigation, traffic management, and pollution control. The enabling techniques were compared with other methods in real-time. Further, the system was analyzed in the light of existing IoT infrastructure.

An Intelligent traffic monitoring system (ITMS) based on the Internet of Things (IoT) was proposed by Putra and Warnars (2018). The proposed system utilized the following four sensors namely, Motion Sensor, Ultrasonic Sensor, Passive Infra-Red (PIR) Sensor and Speed Sensor. The proposed intelligent traffic monitoring system (ITMS) has four components: Arduino, GSM, sensors, and LCD screen. The proposed system ensured that motor vehicles were driven to their destinations, as well as reduced the impact of accidents.

Putra and Warnars (2018) proposed an Intelligent Traffic Monitoring System (ITMS) architecture where the ITMS measures the speed of the driver until the destination using motion sensors and ultrasound sensors. This is used to know when he beats the traffic light.

Cowdrey and Malekian (2018) designed a system to open the security gate of a house by first of all recognizing the vehicular license plate using Optical Character Recognition OCR on license plate characters based on Artificial Neural Network. They used a proximity sensor to tell the camera when the vehicle is in range enough for image capture to commence. An ambient light sensor control circuit is used to control the lighting condition of the environment to help the camera capture better images for the OCR process. Their system is also able to produce SMS notifications regarding security gate access attempts. This demonstrates the fact that this concept is tenable in an intelligent traffic management system.

Sridhar et al. (2019) provided Internet of Vehicles (IOV) based on intelligent traffic management system. The IOV is based on the internet, detection technologies and network wireless sensors to recognize traffic object, monitoring, managing and tracking & processed automatically. The proposed system is mainly based on the concept of Internet of Things (IOT). The basic functionalities of the proposed system include monitoring of speed limits, pollution checks, and emergency response to road accidents and providing security to the server.

The authors Khattak et al. (2019) proposed a novel framework for architectural and communication design to effectively integrate vehicular networking clouds with IoT, referred to as VCoT, to materialize new applications that provision various IoT services through vehicular clouds. They also presented deep insights for different real-world application scenarios (i.e., smart homes, intelligent traffic light, and smart city) using VCoT for general control and automation along with their associated challenges. Finally, they presented initial insights, through preliminary results, regarding data and resource management in IoT-based resource constrained environments through vehicular clouds.

Muthuramalingam et al. (2018) proposed a case study on Intelligent Traffic Management System based on IoT and big data, which will be a part of, smart traffic solutions for smarter cities. The ITS-IoT system itself forms an eco-system comprising of sensor systems, monitoring system and display system. The proposed case study will examine and explain a complete design and implementation of a typical IoT-ITS system for a smart city. How concepts like Multiple regression analysis, Multiple discriminant analysis and logistic regression, Cojoint analysis, Cluster analysis and other big data analytics techniques will merge with IoT and help to build IoT-ITS will also be emphasized. The case study will also display some big data analytics results and how the results are utilized in smart transportation systems.

Saha et al. (2021) proposed a traffic-aware quality-of-service (QoS) routing scheme in software-defined internet of things (SDIoT) network. The proposed scheme exploits the unique features of software-defined networking (SDN), such as flow-based nature, and network flexibility, in order to fulfill QoS requirements of each flow in the network. They considered two types of QoS routing strategies—delay-sensitive and loss-sensitive—for incoming packets from end-devices in the network. The former was devised to deal with delay-sensitive flows, and the latter deals with loss-sensitive flows, in order to maximize the overall network performance. They also proposed a greedy approach based on Yen's

K-shortest paths algorithm to compute the optimal forwarding path, while considering the QoS requirements of each packet. Consequently, the SDN controller deploys adequate flow-rules at the forwarding devices in the network. Extensive simulation results show that the proposed scheme significantly reduces the end-to-end delay and the percentage of flows which violate QoS constraints compared to the benchmarks considered in the study. It was also observed that the proposed scheme adequately satisfies the QoS requirements for both type of flows in contrast to the existing schemes. In particular, with 2000 flows in the network, the proposed scheme achieved 13%, 14% and 15% (with AttMpls topology) and 38%, 37% and 39% (with Goodnet topology) reduction in QoS violated flows as compared to the existing LARAC, SPD, and MRC schemes, respectively.

Rodríguez et al. (2019) proposed an architecture combining the NFV and SDN concepts to provide the logic for Quality of Service (QoS) traffic detection and the logic for QoS management in next-generation mobile networks. In order to manage traffic with QoS requirements, the work incorporated Multiprotocol Label Switching (MPLS) in the mobile data plane. A new flexible and programmable method to detect traffic with QoS requirements was also proposed, along with an Evolved Packet System (EPS)-bearer/QoS-flow creation with QoS considering all elements in the path. The goals were achieved by using proactive and reactive path setup methods to route the traffic immediately and simultaneously process it in the search for QoS requirements. Finally, a prototype was presented to prove the benefits and the viability of the proposed concepts.

Bagaa et al. (2020) concentrated on reducing the operating expenditure (OPEX) costs while increasing the quality of service (QoS) by leveraging the benefits of queuing and multi-path forwarding in OpenFlow, allowing an operator with an SDN-enabled network to efficiently allocate the network resources considering mobility, and reducing or even eliminating the need for over-provisioning. For achieving these objectives, a QoS aware network configuration and multipath forwarding approach was introduced that efficiently manages the operation of SDN enabled open virtual switches (OVSs). They also proposed and evaluated three solutions that exploit the strength of QoS aware routing using multiple paths. While the first two solutions provided optimal and approximate optimal configurations, respectively, using linear integer programming optimization, the third one was a heuristic that uses Dijkstra short-path algorithm. The obtained results demonstrate the performance of the proposed solutions in terms of OPEX and execution time.

An efficient routing scheme, QROUTE, for satisfying multiple QoS constraints in software-defined overlay networks was proposed by Varyani et al. (2020). QROUTE consists of a control plane routing algorithm which has significantly low route computation time because of employing a novel directed-acyclic-graph (DAG) based approach. QROUTE also reduces the forwarding entries in the data plane by using a QoS-metric-based forwarding scheme. The authors extensively evaluated QROUTE using traces from a global overlay service provider. They also examined QROUTE on a testbed of P4-BMv2 switches controlled by the ONOS controller using P4Runtime protocol. The results obtained show that QROUTE outperforms other state-of-the-art QoS routing schemes in route computation time, size of the forwarding tables and meeting the QoS requirements of various applications.

Mubarak et al. (2019) discussed actual task of ensuring the quality of services in information networks with fractal traffic. The generalized approach to traffic management and quality of service based on the account of multifractal properties of the network traffic was proposed. To describe the multi-fractal traffic properties, it was proposed to use the Hurst exponent, the range of generalized Hurst exponent and coefficient of variation. Methods of preventing of network overload in communication node, routing cost calculation and load balancing, which based on fractal properties of traffic were also presented. The results of simulation have shown that the joint use of the proposed methods can significantly improve the quality-of-service network.

Yang et al. (2020) addressed two problems related to the new Segment Routing (SR) mechanism: enabling fine-grained end-to-end QoS routing under a complex network environment and constructing the multicast routing tree with branch node load balancing. To solve these problems, an Inaccurate information-based QoE-driven Routing algorithm

(IQdR) and a Branch-aware Multicast Tree (BaMT) algorithm were proposed. Simulation test results that compared the performance of the proposed solution against that of other algorithms show that the previous works were outperformed. Additionally, the results also show that the multicast architecture improves the scalability of the network in terms of the number of flows.

A novel multicasting technique to guarantee QoS for multimedia applications over SDN was proposed in (Mohammadi et al., 2018). To deliver multimedia contents in an efficient manner, the proposed method models multicast routing as a delay constraint least cost (DCLC) problem. As DCLC problem is NP-Complete, an approximation algorithm using teaching–learning-based optimization to solve the problem was proposed. Experimental results confirmed that the proposed method outperforms IP multicast routing protocol, and it achieves a gain of about 25% for peak signal-to-noise ratio.

2.2.3 Related Works on Fuzzy Logic

Ali (2020) proposed coordinated control method for an arterial road network. The proposed method is based on fuzzy logic and Webster optimum cycle formula. It is a cyclic method, which means that all-feasible phases at the intersection will get at least a minimum green signal during each cycle. These minimum green times can be used for pedestrian crossing purposes. The method eliminated the starvation that occurs at minor roads due to the non-cyclic strategy. The proposed method was investigated in both coordinated and isolated circumstances. It was compared with non-optimized fixed time control and the cyclic backpressure strategy. Based on the obtained results, the adaptive fuzzy logic and Webster based coordinated method outperforms the other methods in terms of the average of waiting time, travel time, travel speed, and queue lengths.

Garg and Kaushal (2017) designed a traffic lights control system for Indian cities using WSN and fuzzy control. Multiple fuzzy logic controllers were designed, one for each phase, that work in parallel. Each fuzzy controller addresses vehicles turning movements and dynamically manages both the phase and the green time of traffic lights. The proposed system combined the typical advantages of WSNs such as, easy deployment and maintenance, flexibility, low cost, noninvasiveness, and scalability with the benefits of using four parallel fuzzy controllers, one for each phase, instead of a single controller for all the phases which resulted to better performance, fault-tolerance, and support for phase-specific management. Simulation results demonstrated that the multi-controller approach here proposed outperforms related works in terms of reduction of the vehicle waiting times in the queues, especially under heavy traffic.

The authors Acharya et al. (2019) proposed using fuzzy logic technique in the field of traffic control system. Electronic sensors were used to detect number of vehicles waiting at the traffic junction and hence action can be taken accordingly to control the traffic jam. The work demonstrated that fuzzy logic controller system is more effective and has better performance over conventional controller system with cost effect in the field of decision making to control the traffic.

The authors Kukic and Jovanovic (2019). developed a model for controlling a diverging diamond interchange – a type of diamond interchange in which the two directions of traffic on the non-freeway road cross to the opposite side on both sides of the bridge at the freeway. The model has been implemented in many different locations, above all in the USA.

An intelligent density traffic control system using (fuzzy logic) which is capable of providing priority to the road users based on the density and emergency situations was developed and presented in (Adewale, 2018). The system will obtain the approximate amount of vehicle and presence of pedestrians respectfully on each lane with help of Infrared Sensors (IR) and siren detection system for emergency and security road users. The working principle of the system depending on the logic inputs rules given into the processing unit by the (sensors, S1 and S2) helps the system to generates a timing sequence that best suit the number of vehicles and pedestrians available on the lane at point in time.

The author in (describe the design and implementation of a conventional traffic light-controlled technique along with the concept of fuzzy logic technology. Conventional traffic controllers use time-sharing principle for controlling directions of vehicles. A traffic light controller primarily based on fuzzy logic was used for optimum control of fluctuation traffic volumes consisting of over saturated or unusual load situations. The authors controlled the working of the traffic lights based on the traffic density on the lanes, but also, we will consider emergency cases to manage its working. The emergency cases include: arrival of an ambulance or any delegate's or minister's vehicle.

Ikidid et al. (2021) presented a traffic optimization system based on agent technology and fuzzy logic that aims to manage road traffic, prioritize emergency vehicles, and promote collective modes of transport in smart cities. The approach aimed to optimize traffic light control at a signalized intersection by acting on the length and order of traffic light phases in order to favor priority flows and fluidize traffic at an isolated intersection and for the whole multi-intersection network, through both inter- and intra-intersection collaboration and coordination. Regulation and prioritization decisions are made on real-time monitoring through cooperation, communication, and coordination between decentralized agents. The performance of the proposed system was investigated by implementing it in the AnyLogic simulator, using a section of the road network that contains priority links. The results indicate that the system can significantly increase the efficiency of the traffic regulation system.

Daeichian and Haghani (2019) introduced a multi-agent approach to adjust traffic lights based on traffic situation in order to reduce average delay time. In the traffic model, lights of each intersection were controlled by an autonomous agent. Since decision of each agent affects neighbor agents, the approach created a classical non-stationary environment. Thus, each agent not only needs to learn from the past experience but also has to consider decision of neighbors to overcome dynamic changes of the traffic.

Karmakar et al. (2021) introduced an Emergency Vehicle Priority System (EVPS) by determining the priority level of an EV based on the type and the severity of an incident, and estimating the number of necessary signal interventions while considering the impact of those interventions on the traffic in the roads surrounding the EV's travel path. They presented how EVPS determines the priority code and a new algorithm to estimate the number of green signal interventions to attain the quickest incident response while concomitantly reducing impact on others. A simulation model was developed in Simulation of Urban Mobility (SUMO) using the real traffic data of Melbourne, Australia, captured by various sensors. Results show that the system recommends appropriate number of interventions that can reduce emergency response time significantly.

Wang et al. (2018) investigated the development of commonly used self-adaptive signal control systems in the world, their technical characteristics, the current research status of self-adaptive control methods, and the signal control methods for heterogeneous traffic flow composed of connected vehicles and autonomous vehicles. Finally, they concluded that signal control based on multiagent reinforcement learning is a kind of closed-loop feedback adaptive control method, which outperforms many counterparts in terms of real-time characteristic, accuracy, and self-learning and therefore will be an important research focus of control method in future due to the property of “model-free” and “self-learning” that well accommodates the abundance of traffic information data. Therefore, the related achievements of the adaptive control system for the future traffic environment have extremely broad application prospects.

In (Deveci et al., 2020), in order to capture individual and multiple opinions from decision makers over repeated surveys, the Interval Agreement Approach (IAA) was utilized to prioritize and evaluate the smart city dimensions for Istanbul's recent smart city project. The validity of the suggested approach was tested with a case study of Istanbul, the largest city in Turkey with many urban problems such as chronic traffic gridlock, rapid urban growth and environmental pollution. The results from the experiments show that IAA can be a useful approach for researchers who theoretically study smart city dimensions, concepts and decision-making methods. It can also help practitioners who deal with implementing smart city projects.

Vogel et al. (2018) presented an adaptive traffic light controller based on fuzzy logic for improving the traffic flow on an isolated intersection. A set of fuzzy rules was made, such that using the collected information from road detectors (queue length, arrival flow, and exit flow), it computed the amount of time for which the next phase should be shortened or extended. The proposed fuzzy control system constitutes of two parts: one for the primary driveway (with a higher volume of vehicles) and the other for the secondary driveway (with a lower volume of vehicles). The proposed controller was compared with a fixed signal program in three scenarios with different traffic demand proofing the effectiveness of the developed decision rules.

Adak and Balta (2021) performed different traffic scenarios based on VANET on SUMO for adaptive and non-adaptive intersections. The gathered traffic information data from vehicles were given to the developed fuzzy logic model to optimize green light durations. A Period of a scenario for analyzing took 10 minutes, and according to 10 minutes input, the fuzzy model optimizes the green light durations for the following period. Test results show that using a fuzzy model in traffic light optimization decreases the average waiting time of vehicles and average queue length.

Ariffin et al. (2021) proposed an intelligent system that dynamically adjusts the cycle length for each of the lanes at an intersection based on the vehicular density and grants the different types of EVs to pass through the intersection point by assigning different priorities. The system consists of three modules, the traffic light control module, the emergency Radio-frequency identification (RFID) module and the internet module. The traffic light control module detects vehicular density using ultrasonic sensors and assigns a dynamic set of cycle length based on the individual lane density condition. The emergency RFIDs are installed on different types of EVs based on the preset priority weights. The internet module allows the dynamic traffic light system to be controlled by authorized personnel in a real-time application.

Alomari et al. (2020) proposed an ACC using a Fuzzy Logic approach for an autonomous model car called “AutoMiny.” AutoMiny was developed at the Dahlem Center for Machine Learning and Robotics at Freie Universität Berlin. It navigates by correcting its orientation error given by a global localization system and a pre-built grid map. The proposed controller can handle two states with differently designed profiles.

George, et al. (2018) utilized Internet of Things (IoT) and Adaptive Neuro Fuzzy Inference System (ANFIS) to improve traffic conditions. An ANFIS traffic light controller with inputs as waiting time and vehicle density was developed using MATLAB SIMULINK environment. A camera was used to capture the traffic scenes and the image was transferred to the cloud using Arduino UNO and ThingSpeak Platform. The image was then analyzed in the server using ANFIS controller and appropriate control signals were sent to the traffic signals. Experimental results showed that the approach performed well for a simple traffic light intersection scenario.

In (Deveci et al., 2021), the advantages of six different real-time traffic management methods, which are dynamic speed limits, lane control systems, variable message signs, ramp metering, traffic diversion, and integrating autonomous vehicles into other traffic management systems, with four main criteria, namely economic, public and political, environmental, and traffic safety, and 13 sub-criteria using fuzzy multi-criteria decision making (MCDM) were prioritized. The authors proposed novel extensions of the combined compromise solution (CoCoSo) methodology, including the logarithmic method and the Power Heronian function. The validation of the results was presented to verify the flexibility of the proposed methodology. Finally, these results of the effect of parameters p and q on decision outcome show that the ranking results were not influenced.

Latif and Megantoro (2020) presented a simulation control system which can perform automatic traffic light control based on vehicle arrival rate. The method used to control the flame duration is fuzzy logic algorithm with the Mamdani fuzzy reasoning using Matlab software. Test results based on the simulation of fuzzy logic in fuzzy MatLab toolbox show that the fuzzy logic algorithm can be used to meet the goal of optimal traffic control, the duration of a given time based on the arrival rate of vehicles. The higher rate of arrival of the vehicle, the longer the duration of a given time, and the lower rate of the arrival of the vehicle, the more briefly given time duration.

Ikidid and Abdelaziz (2019) proposed a traffic simulation framework based on agent technology and fuzzy logic. The objective of the framework was to act on the duration of traffic lights to fluidize traffic in an insulating intersection in collaboration with neighboring intersections and in an intersection network. The calculation of light durations was done in real time based on the flow and concentration of traffic on the sections to reduce the travel time. The light regulation was achieved through communications and collaborations among agents.

2.2.4 Related Works on Artificial Neural Network

Louati et al. (2021) investigated the potential of using convolution neural network (CNN) in detecting emergency cases and forecasting events that can interrupt the traffic flow. Case-based reasoning (CBR) was then exploited to react to detected and forecasted events. They further developed a Reinforcement Leaning (RL) algorithm in building and enhancing the case bases. The proposed system inherited the advantages of CNN, CBR, and RL, which allow detection, prediction, control, evaluation, and learning in a unified framework. To assess the proposed TSCS, the authors compared the approach with a set of state-of-art algorithms (e.g., multi-agent preemptive case-based reasoning algorithm and multi-agent preemptive longest queue first—maximal weight matching). The proposed TSCS outperforms the benchmarking algorithms through experiments in various traffic scenarios.

Sahraei (2018) sampled data from eight priority junctions of various configurations. Data pertaining to the analysis of critical gap, follow-up time, and control delay were collected using video camera recording technique. The study was divided into two phases comprising analysis of field data, and the development of ANN and mathematical models using MATLAB software. In the course of data analysis, the research recognized and estimated various variables that influence control delay. To generate the model, an ANN with two hidden layers and several sizes of neurons in the hidden layers were developed. Several mathematical models for estimation of control delay with a reasonable accuracy were developed using the outputs from the ANN model. Findings from this research showed that the range of conflicting flow is from 130 to 2470 veh/h and 120 to 2300 pcu/h, the values of control delays predicted are 3-37 sec/veh and 4-43 sec/pcu, respectively. The modelling results showed that the values of control delay for right-turning maneuvered from minor road at junction with four lanes major/two lanes minor road were higher than other junctions.

Africa (2019) carried out research on a traffic control system using sensors and a neural network. It utilized vision-based sensors to monitor intersection congestion data and sends this data to the surrounding stoplights to optimize traffic flow. The neural network was trained to intercept the data collected in each stoplight and control the stoplight signals to direct the cars in the most efficient way possible. The neural net was trained via simulation and optimized based on the average travel time of each simulated vehicle to rate its performance.

This study by Celtek (2021) proposed a novel approach to the Deep Q-Learning based adaptive traffic control system to determine the best action. The approach did not just aim to minimize delay time by waiting time during the red-light signal but also aimed to decrease delay time caused by vehicles slowing down when approaching the intersection, and the delay time caused by the required time to accelerate after the green light signal. The performances of the methods were evaluated in real-time through the Simulation of Urban Mobility traffic simulator. The results clearly show that proposed method decreases the delay time.

Chen et al. (2020) evaluated the use of various smoothing models for cleaning anomaly in traffic flow data, which were further processed to predict short term traffic flow evolution with artificial neural network. The wavelet filter, moving average model, and Butterworth filter were carefully tested to smooth the collected loop detector data. Then, the artificial neural network was introduced to predict traffic flow at different time spans, which were quantitatively analyzed with commonly-used evaluation metrics. The findings of the study provided efficient and accurate denoising approaches for short term traffic flow prediction.

Saadaoui et al. (2019) developed a hybrid approach coupling feedforward neural networks with a nonlinear least squares-based regression curve fitting for the multistep-ahead prediction. Empirical experiments were conducted in order to demonstrate the effectiveness of the proposed model on passenger traffic real datasets. The results showed that the base model is capable of generating accurate forecasts, with a performance comparable with that of powerful state-of-the-art forecasting models.

Sharma et al. (2018) developed a short-term traffic forecasting model using back propagation artificial neural network for two lane undivided highway with mixed traffic conditions. The results were compared with random forest, support vector machine, k-nearest neighbor classifier, regression tree and multiple regression models. It was found that back-propagation neural network performs better than other approaches and achieved an R2 value 0.9962, which is a good score.

In (Ata et al., 2019), the prediction of congestion was operationalized by using the algorithm of backpropagation to train the neural network. The proposed system aimed to provide a solution that will increase the comfort level of travelers to make intelligent and better transportation decision, and the neural network is a plausible approach to find traffic situations. Proposed MSR2C-ABPNN with Time series gave attractive results concerning MSE as compared to the fitting approach.

A study was carried out to the modeling of control delays at unsignalized intersection using Artificial Neural Network (ANN) by Sahraei and Puan (2018). Data pertaining to the analysis of control delay was collected from three intersections of various configurations using video camera recording technique. An ANN with two hidden layers and several sizes of neurons in the hidden layers were developed. Two mathematical models for estimation of control delay from minor road with a reasonable accuracy were developed using the outputs from the ANN's model. The results of this research showed that the neural network is able to predict control delay incurred on minor road vehicles at unsignalized intersection more accurately. The analysis revealed that heavy vehicles had the lowest effect on the proposed formulas, where by increasing from 10% to 50%, the values of control delay could increase from 1% to 3%, while the movement flow and conflicting flow had the highest impact, where within the same ranges; control delay could increase until 39%.

Chen et al. (2020) proposed an ensemble framework via ensemble empirical mode decomposition (EEMD) and artificial neural network (ANN) to predict traffic flow under different time intervals ahead to help traffic participants collect more accurate traffic flow data and benefits transportation practitioners by helping them to make more reasonable traffic decisions. More specifically, the proposed framework firstly employed the EEMD model to suppress the noises in the raw traffic data, which were then processed to predict traffic flow at time steps under different time scales (i.e., 1, 2, and 10 min).

Sopena et al. (2021) presented a hybrid traffic prediction framework producing both point and interval forecasts utilising a combination of mode decomposition algorithms along with Artificial Neural Networks (ANN). A new decomposition algorithm known as Variational Mode Decomposition (VMD) was introduced to decompose traffic into a set of subseries which are modelled using ANNs. A quantile regression loss function was implemented to estimate prediction intervals. The performance of the proposed approach was evaluated using traffic flow data collected from a signalized junction in Dublin City (Ireland) and compared against a set of alternative hybrid models. Furthermore, the robustness of the proposed algorithm was established through consistency in performance when tested over varied prediction horizons.

Viloria et al. (2020) proposed a solution to the problem of vehicle control through a proactive approach based on Machine Learning. Through the solution, a traffic control system learns about traffic flow in order to prevent future problems of long queues at traffic lights. The architecture of the traffic system was based on the principles of Autonomous Computing with the aim of changing the traffic light timers automatically. A simulation of the roads in an intelligent city and a Weka-based tool were created to validate the approach.

Nguyen et al. (2018) reviewed recent studies of deep learning for popular topics in processing traffic data including transportation network representation, traffic flow forecasting, traffic signal control, automatic vehicle detection, traffic incident processing, travel demand prediction, autonomous driving and driver behaviors. In general, the use of deep learning systems in transportation is still limited and there are potential limitations for utilizing this advanced approach to improve prediction models.

In order to improve the operation efficiency of the intelligent traffic control system, based on the open Internet of Things and machine learning, Chen and Liu (2020) built an intelligent three-way intelligent traffic control system, as well as a simulation model using cellular automata as a platform. In addition, they analyzed the developed model through the statistics of the highest vehicle flow on the road and the relationship between road occupancy and vehicle speed. The research results show that the developed model has good performance and can be applied to intelligent traffic control.

Gedik (2020) developed a system to predict the short-term traffic volume of a connection road leading to one of Istanbul's Bosphorous Bridge in Turkey by the three different implementations of ANN. These were Feed Forward Back Propagation (FFBP), Generalized Regression Neural Network (GRNN) and Radial Based Function (RBF). Then, obtained results were compared with each other and the result of Multi Linear Regression (MLR) method.

A seasonal autoregressive integrated moving average plus seasonal discrete grey model structure (SARIMA-SDGM) was proposed to perform the traffic speed prediction by Song et al. (2019). The model performance of SARIMA-SDGM model was compared with that of the seasonal autoregressive integrated moving average (SARIMA) model, seasonal discrete grey model (SDGM), artificial neural network (ANN) model, and support vector regression (SVR) model. The results showed that SARIMA-SDGM model performs best with the lowest mean absolute error (MAE), mean absolute percentage error (MAPE), and the root mean square error (RMSE). The traffic speed prediction accuracy under different time intervals were compared based on the SARIMA-SDGM model. The results showed that the prediction accuracy improves with the increase in time interval. In addition, when the time interval is greater than 10 min, the prediction results yield stable prediction accuracy.

In (Sumalatha et al., 2019), an attempt was made to model and forecast short-term traffic flow by applying Neural Network models. The traffic data was considered for peak hours in the morning for 8:00am to 12 Noon, for 5 days. Multilayer Perceptron (MLP) network model was used in the study. The results can be considered to monitor traffic signals and explore methods to avoid congestion at that junction.

In (Deulkar, 2019), an ANN simulation method was used to analyze the sensitivity of toll plaza performance for different types of traffic flow. Two traffic patterns, deterministic and probabilistic traffic flow, were considered. The study was based on a proposed project for building a toll Plaza in the indirectly. The estimated future traffic counts for the toll bridge were used to study the difference between the two traffic patterns.

Awoyera et al. (2019) presented a novel traffic control schemeITCSwith machine learning abilities. The Intelligent Traffic Control System (ITCS) consists of Closed-circuit television (CCTV) cameras that take photograph of each traffic lane in real time, and send to the Image Processing unit which determines the volume of traffic on that particular lane. The ITCS then assigns a priority to each lane based on the current traffic volume on it. The priority weights can be adapted in real time, and are capable of responding to traffic changes caused by unforeseen events. The Adaptive Neuro-Fuzzy Inference System (ANFIS)-based traffic control system can learn from past traffic data and can predict future traffic on a particular road, by observing the traffic on the adjoining roads.

2.3 Research Gaps

The following research gaps were carefully identified after a systematic review of the literature as presented in the previous sections.

1. Non-application of DTAT algorithm leveraging Fuzzy Logic and ANN for intelligent traffic management system.
2. Non-utilization of inexpensive and off-the-shelf materials in the development of automatic traffic monitoring and control system.
3. Non-application of load balancing algorithm (a distributed computing paradigm) for the control of road traffic signals.
4. Lack of a traffic monitoring and control system with the integrated capabilities of:

i. Reducing average waiting time of vehicles,
ii. Capturing license plate number of the traffic offender,
iii. Communicating traffic offender via SMS gateway on the exert fines and fees for his traffic offence,
iv. Reporting defaulters of traffic rules and regulations to appropriate authorities.

CHAPTER THREE
MATERIALS AND METHOD

3.1 Materials

The materials utilized in this work are:

i. Motion Sensors
ii. Software Analysis and Modelling Tool
iii. System Simulation Tool
iv. GSM Modules
v. Firmware Development Tools
vi. LEDs
vii. Arduino Microcontrollers
viii. IP Cameras
ix. Mobile Communication Gadgets

3.2 Method

In this work, a composite methodology has been adopted. It involves:

i. Object-Oriented System analysis and design utilizing Unified Model Language (UML) tools.
ii. Dynamic Time Allocation development techniques as well as Fuzzy logic and Artificial Neural Network (ANN) design techniques.

3.2.1 System Analysis and Modeling

The analysis and modeling of the developed Intelligent Traffic Management system is discussed in sub-section 3.2.1. This will be accomplished by carrying out object-oriented analysis of the proposed system, and modeling it using UML tools.

3.2.1.1 Object-Oriented Analysis of Intelligent Traffic Management System

This includes information gathering, functional requirements, and non-functional requirements of the system as well as system description.

A. Requirements Gathering

In an effort to understand the principles of operation of traffic signals and how they function in Owerri Metropolis, an interview was conducted at the Federal Road Safety Corp (FRSC), on the 15th November, 2021. Discussions held disclosed that there is indeed a need for a traffic signal system that will emulate the behaviour of a human traffic controller, thereby automatically monitoring and controlling road traffic conditions. It was then added that this will reduce the workload of traffic marshals and the risk of road traffic accidents and death, as well as facilitating the effective and efficient enforcement of laws associated with road usage and safety. It was also in this interview that request was made to physically observe traffic flow at junctions and also watch and interact with road traffic marshals who monitor and control traffic. Consequently, this was granted.

Interactions with various traffic monitoring and control officers as well as drivers at different locations revealed that traffic is usually dense during rush hours of the mornings (between 7am – 10am) and evenings (between 5pm – 8pm). However, the possible causes of dense road traffic include disobedience of traffic rules by vehicle drivers, breakdown of vehicles (especially heavy-duty) on the roads, numerous vehicle drivers contending for a particular route, and miscalculations on the part of the officers on how to strike optimal balance of traffic flow in different lanes of the road; these constitute a major challenge, the officials and drivers added.

The above events led to the understanding of the operations of the already existing traffic as thus;

i. The change of state of the traffic signal from red through amber to green (and vice versa) is based on a digital timer which is calibrated to read from 0 to 60 seconds before effecting the change.
ii. Most times, there are always longer traffic queues on one road lane and another lane virtually free from traffic. This is especially during rush hours of the

mornings (when people are going to work) and evenings (when people are returning from work).

iii. Drivers oftentimes violate road traffic rules which results to deadlock in most cases.

iv. Emergency vehicles such as ambulance, fire truck, etc. find it very difficult to maneuver traffic queues as to get to their destination at the required time.

v. For road junctions that do not have traffic control signals, traffic marshals find it very stressful and challenging to monitor and control road traffic, especially on a rainy day or very sunny day.

This led to the capture of the following requirements of the system, considering the limitations of the already existing traffic signal.

B. Functional Requirements of the System

i. The system will monitor and control road traffic in real-time.

ii. The system will achieve monitoring by capturing the situation of the road in real-time.

iii. The system will achieve control by accepting and processing the captured data regarding situation of the road in real-time.

iv. The system shall store road traffic data in a central database where certain users will have access to certain information.

v. The system will register every vehicle within a specific range of distance which is assumed to be the traffic queue.

vi. The system will unregister every car that exits the traffic queue distance range.

vii. The system will reduce average waiting time of vehicles by striking a balance between the flows of traffic in different lanes.

viii. The system should be able to predict and report traffic situation to drivers so they can make informed decisions.

ix. The system should be able to track vehicles that are reported stolen by appropriate authorities.

x. The system should be able to clear the traffic path of emergency vehicles in order to optimally reduce their travel time.

xi. The system will report defaulters of traffic rules to appropriate authorities.

Requirements Definition

This section defines each of the functional requirements of the system in order to clearly understand their roles and operations with respect to the system.

i. *The system will monitor and control road traffic in real-time*: this is the major functionality of the system which is achieved as outlined in 3.2.1.1 B (ii and iii).

ii. *The system will achieve monitoring by capturing the situation of the road in real-time:* the traffic situation of the road shall be captured by;

   a. Registering every vehicle within the distance range of the traffic queue,
   b. Recording the number of vehicles within the distance range of the traffic queue,
   c. Recording the average waiting time of vehicles on the traffic queue within the distance range at a particular phase of the traffic signal.
   d. Recording the rate at which vehicles enter and leave the traffic zone (i.e., distance between the road junction where the traffic signal is positioned, and the supposed point where the traffic stops) at a particular phase of the traffic signal.
   e. Unregistering vehicles that exit the traffic queue.

iii. *The system will achieve control by accepting and processing captured data regarding the situation of the road:* the road traffic situation is primarily determined by the average waiting time of vehicles on traffic queues which is determined by the number of vehicles in the traffic zone and the rate at which they enter and leave the zone.

The traffic is high if the number of vehicles in the traffic queue is high; the average awaiting time of vehicles on the traffic zone is high; and the average frequency at which vehicles enter and leave the traffic zone is low.

Thus, the road traffic situation, T_s is expressed in equation 3.1 as;

$$T_s = \frac{N_v + T_{avg} + F_{avg}}{T_{ph}} \quad (3.1)$$

Where;

N_v = Number of vehicles in the traffic queue (traffic zone),

T_{avg} = average waiting time of vehicles in the traffic zone,

T_{ph} = traffic phase, and

F_{avg} = average frequency at which vehicles enter and leave the traffic zone at a particular phase of the traffic signal.

The average frequency at which vehicles enter and leave the traffic zone at a particular phase of the traffic signal, F_{avg} is determined using equation 3.2;

$$F_{avg} = \frac{1}{T_{avg}} \quad (3.2)$$

T_{avg} is defined as time taken for the respective vehicles in traffic zone to exit the zone divided by the number of vehicles that left.

Mathematically,

$$T_{avg} = \frac{(t_{ex} - t_{en})_1 + (t_{ex} - t_{en})_2 + \cdots (t_{ex} - t_{en})_n}{n} \quad (3.3)$$

Where;

t_{en} = time of entrance of each vehicle,

t_{ex} = time of exit of each vehicle, and

n = number of cars that leave the traffic zone at a particular phase.

Traffic Phase, T_{ph}**:** at this point, it is important to mention that a traffic phase, T_{ph} in this context refers to the cycle of time in which the traffic signal turns from green light through amber to red for a particular lane. Thus, it is the duration of time in which a particular lane is allowed to flow before being stopped by the red light.

Mathematically,

$$T_{ph} = t_g + t_a \tag{3.4}$$

Where;

t_g = time that the green light of a particular lane is turned on,

t_a = time that the amber light of a particular lane is turned on.

iv. <u>*The system shall store traffic data in a central database where certain users will have access to certain information:*</u> these traffic data include the following;
    a. The real-time traffic situation of the road; which describes whether there is high or low congestion at a particular point in time,
    b. Records of vehicles that have violated traffic rules with the time, date, location, and a snapshot of the event.
    c. Forecast of traffic of the day and week.
    d. Record of vehicles that have been reported stolen.

v. <u>*The system will register every vehicle within a specific range of distance which is assumed to be the traffic queue:*</u> each car that enters the traffic zone (say 400 meters from the traffic signal) will be registered by the system in order to ascertain the number of vehicles and their average waiting time on a traffic queue within a specific period of time.

vi. <u>*The system will unregister every car that exits the traffic queue distance range:*</u> each car that leaves the traffic queue (traffic zone) shall be unregistered by the system in order to create room for new cars to be registered and to ascertain the traffic situation T_s.

vii. <u>*The system will reduce average waiting time of vehicles by striking a balance between the flow of traffic in different lanes:*</u> this will depend on the traffic situation of the road. If there is equal amount of traffic in all the lanes of the road, the system will make use of a balanced clocking signal (load balancing) for the traffic control. Else, the system will control the traffic based on sensed situation of the road as per lanes with higher and lower traffic queue.

viii. *The system should be able to predict and report traffic situation to drivers so they can make informed decisions:* traffic situation for a particular time of the day or day of the week shall be predicted and stored in a centrally accessible database so that road users can be able to appropriate choice of route to use at a particular point in time.

ix. *The system should be able to track vehicles that are reported stolen by appropriate authorities:* details of vehicles (plate number, colour, type, manufacturer, chassis number, engine number, etc.) that are reported stolen shall be uploaded to a centrally accessible database which will then be stored in the traffic signal. When such vehicle is sensed, it will be captured and its details forwarded to appropriate authorities in real-time. The traffic signal for that particular lane where the stolen vehicle is sensed is turned red until the police (or equivalent) apprehends the vehicle, and overwrites the traffic signal control back to its normal state.

x. *The system should be able to clear the traffic path of emergency vehicles in order to optimally reduce their travel time:* emergency vehicles shall transmit a special emergency code to the traffic signal which will activate its light to green. The emergency vehicle shall be registered and upon passing the junction where the traffic signal is stationed, it becomes unregistered and the traffic signal restored back to its normal state.

xi. *The system will report defaulters of traffic rules to appropriate authorities:* details of vehicles that violate traffic rules shall be captured and stored to a central database where it shall be accessible by appropriate authorities. Upon clearance of the fine, the appropriate authorities shall wipe out the details of the vehicle from the central database.

C. Non-functional requirements

i. The system should not allow a vehicle to wait more than 10 minutes on a traffic queue.

ii. The range of traffic distance covered by the system should be about 400 meters from the road junction where the signal is situated.

iii. The network throughput of the interconnected systems should be optimal in order to facilitate effective and real-time communication and responsiveness of the entire system.

iv. The system should protect the centralized traffic information from unauthorized access to avoid malicious damage.

D. Description of the Intelligent Traffic Management System

This project work develops a sensor array capable of detecting vehicular presence and vehicular movement in an embedded system, and uses this array in building a traffic light system where Dynamic Time Allocation Technology can be implemented. Inductive sensor array and Passive Infra-Red PIR sensor array are used concurrently to increase vehicular sensing capability.

In order to do Dynamic Time Allocation Technology, fuzzy logic is used to act upon the data coming from the sensor. Here, assuming a T-junction having three roads, the traffic situation on each road of the junction constitutes an input into the fuzzy logic algorithm. After following all the fuzzy logic procedures, green time and red time – which is the output of the fuzzy logic system is optimized for each road in the junction.

When a driver beats the traffic light, a sensor will trigger the camera to capture the vehicle license plate number. To do this, image recognition algorithm will come to play. The image from the camera will serve as input to the algorithm. First, the image will be localized for the license plate. Character segmentation will be done on the image in other to extract the characters on the license plate. The segmented characters will go through a character recognition procedure to generate a text output of the characters of the license plate.

With the license plate so gotten, it will be used as the primary key to search for the vehicle owner's phone number from the FRSC database. The database will output the phone number and address of the vehicle owner.

The SMS gateway will then be utilized to send a message to the vehicle owner, telling him of his offence and the fine to be paid due to his offence. The architecture of the proposed system is shown in Figure 3.1.

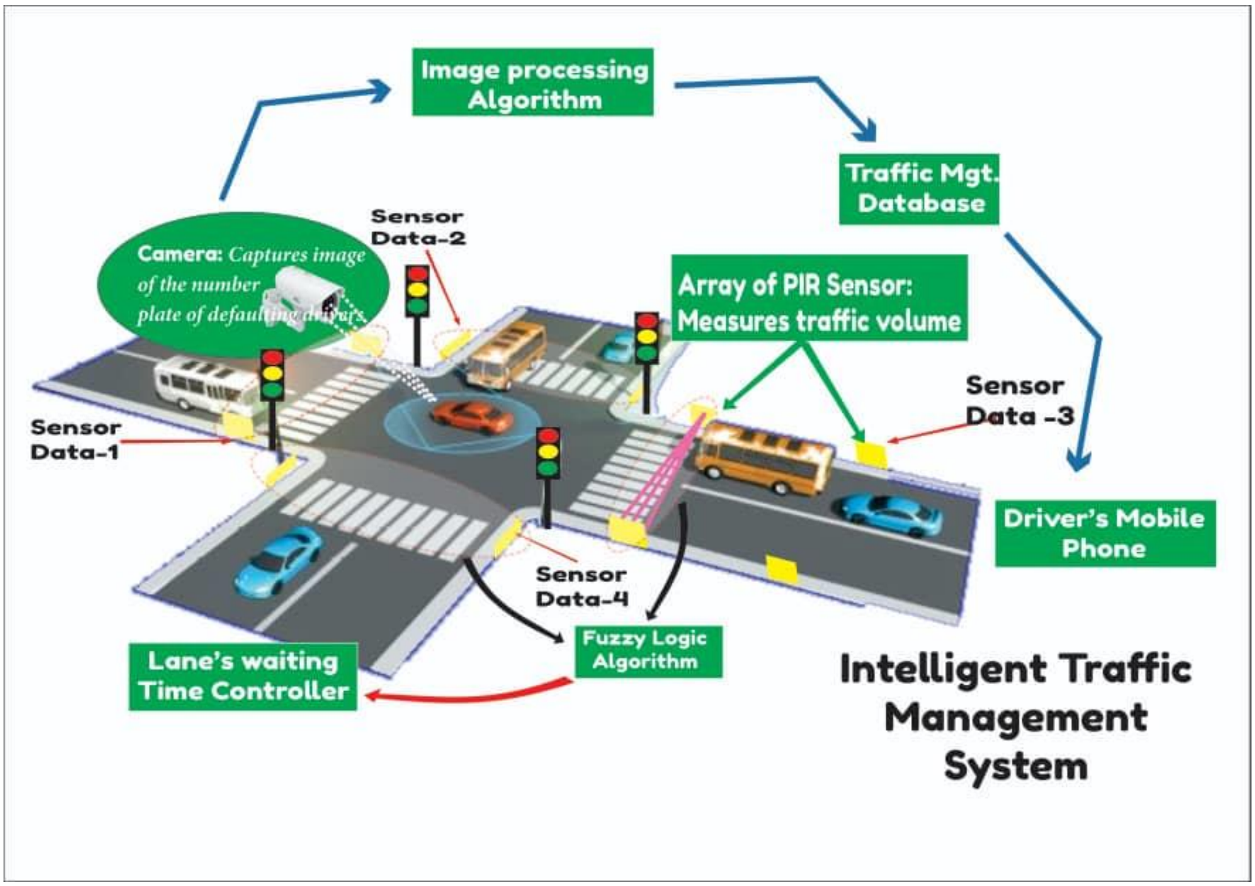

Figure 3.1: An Architecture of the proposed system

3.1.1.2 Modelling of Intelligent Traffic Management System

The developed intelligent traffic management system is modeled using appropriate object-oriented design principles and patterns. The UML representation of the system includes use cases, tabular presentation of major entities, data, and operations as well as conceptual class diagram, use case diagram and activity diagram.

(A) Use Case Analysis

This section details use cases for requirements/operations that are deemed very important and complex to understand. Tables 3.1 to 3.4 depicts the four basic use cases.

TABLE 3.1: USE CASE (1); MONITOR ROAD SITUATION

Use Case Number:	1
Use Case Name:	Monitor Road Situation
Priority:	HIGH
Primary Actor	Digital Proximity Sensor and Traffic Signal
Dependent on:	Road, Vehicles, and Traffic Signal
Brief Description:	This use case describes how the system monitors the traffic situation of a particular lane.
Precondition:	System is functioning properly, appropriate sensors are positioned at appropriate points, and there is no interference in communication between the subsystems.
Trigger	Vehicle enters and leaves the traffic zone.
Main Flow:	
1.1. The system assumes that no vehicle has entered the traffic zone of a particular	

lane; the traffic flow for that lane is free.

1.2. The system observes when a vehicle will enter the traffic zone.

If a vehicle enters the traffic zone:

1.2.1. The system registers the vehicle.

1.2.2. The system records the time of registration.

1.2.3. The system increases the number of vehicles in the traffic zone by 1 i.e., ++1

1.2.4. The system records the total number of vehicles in the traffic zone.

If a vehicle leaves the traffic zone:

1.2.5. The system unregisters the vehicle.

1.2.6. The system records the time of exit.

1.2.7. The system calculates the vehicle waiting time on traffic queue = (exit time – entry time).

1.2.8. The system reduces the number of vehicles in the traffic zone by 1 i.e., --1.

1.2.9. The system records the number of vehicles that exits the traffic zone within a period of time.

Else:

1.2.10. Return to step 1.2.

1.3. The system calculates the frequency at which vehicles enter and leave the traffic zone within a specific time.

If the frequency is HIGH: 1.3.1. Traffic is LOW *If the frequency is LOW* 1.3.2. Traffic is HIGH *If the frequency is NORMAL* 1.3.3. Traffic is NORMAL *Else:* 1.3.4. Return to step 1.3.

Post Condition	The traffic situation of a particular lane of the road has been successfully monitored.
Special Requirements	Digital proximity sensors. Green, yellow, and red LEDs. Short range communication.

TABLE 3.2: USE CASE (2); CONTROL TRAFFIC

Use Case Number	2
Use Case Name	Control Traffic
Priority	HIGH
Primary Actor	Digital Proximity Sensor and Traffic Signal
Dependent on	Monitored road situation (Use case 1)

Brief Description	This use case describes how the system controls traffic based on the sensed road situation.
Precondition	The traffic conditions of all the lanes of the road that meets at the traffic control point have been monitored and understood to be HIGH, LOW, or NORMAL for the respective lane of the road.
Trigger	The traffic condition for the respective lanes of the road has been monitored.
Main Flow:	

2.1 The system assumes all the road lanes have equal traffic flow at default.

2.2 The system picks lanes at random and assigns time for green and red light for traffic flowing and opposing lanes respectively.

2.3 The system takes in monitored traffic reports for each of road lanes.

2.4 The system checks traffic reports for each lane.

If traffic flow for a lane is slower than an opposing lane:

2.4.1. The system increases the wait time of the opposing lane and assigns more flow time to the slower lane.

2.4.2. Return to step 2.4.

Else:

2.4.3. Return to step 2.3.

2.5. The system balances the load of frequency of traffic flow of vehicles for different lanes.

2.6. The system learns on its own the flow of traffic for the specific time of the day.

2.7. The system stores the data in a central database.	
Post condition	Traffic for different lanes of the road has been controlled based on the frequency of flow of vehicles.
Special requirements	ANN framework, and load balancing mechanism. Central database.

TABLE 3.3: USE CASE (3); TRACK STOLEN VEHICLE

Use Case Number	3
Use Case Name	Track Stolen Vehicle
Priority	HIGH
Primary Actor	Police, Central Database, Digital Proximity Sensor, and Traffic Signal.
Dependent on	Stolen vehicle registration on central database, road situation monitoring (use case 1)
Brief Description	This use case details how a stolen vehicle is tracked by the system when in the traffic zone.
Precondition	Stolen vehicle has already been registered on a central database.
Trigger	Reported stolen vehicle enters into traffic zone.
Main Flow:	

3.1 The system fetches list of stolen vehicles from central database. 3.2 The system checks for stolen vehicles as they are registered into the traffic zone. *If a stolen vehicle is found:* 3.2.1 Activate stolen vehicle in traffic zone command. 3.2.2 Turn traffic light for the lane to RED. 3.2.3 Notify the details of the vehicle to the nearest police station. 3.2.4 Keep RED until police comes to overwrite system to default state. *Else:* 3.2.5 Return to step 3.2.	
Post condition	Stolen vehicles have been monitored and captured if found.
Special requirement	Central database. Police.

TABLE 3.4: USE CASE (4) REPORT/PREDICT TRAFFIC SITUATION

Use Case Number	4
Use Case Name	Report Current and Future (Predicted) Traffic Situation
Priority	Medium
Primary Actor	Central database and Road Users.
Dependent on	Already analyzed monitored traffic situation
Brief Description	This use case details how the system predicts and reports traffic situation to vehicle drivers and other entities such as

	traffic radio.
Precondition	The road traffic situation has been monitored for a particular period of time
Trigger	Already analysed road traffic situation within a specific period, road user requests for future traffic situation
Main Flow:	
4.1 Road user requests to know current traffic situation for a particular route. 4.2 System communicates with the database to fetch result of the current traffic situation. 4.3 Road user requests to know traffic situation for future time of the day or day of the week for a particular route. 4.4 System communicates with the database to fetch predicted traffic situation for the requested time of the day or day of the week. 4.5 System communicates traffic situation to the road user. 4.6 Road user makes informed decision based on communicated traffic situation.	
Post condition	Current or predicted traffic situation has been communicated to road user in order to make informed decision.
Special requirement	Fuzzy logic and ANN framework.

(B) Use Case Diagram and Conceptual Class Diagram

The use case diagram for the system as derived from Tables 3.1 to 3.4 is depicted in Figure 3.2.

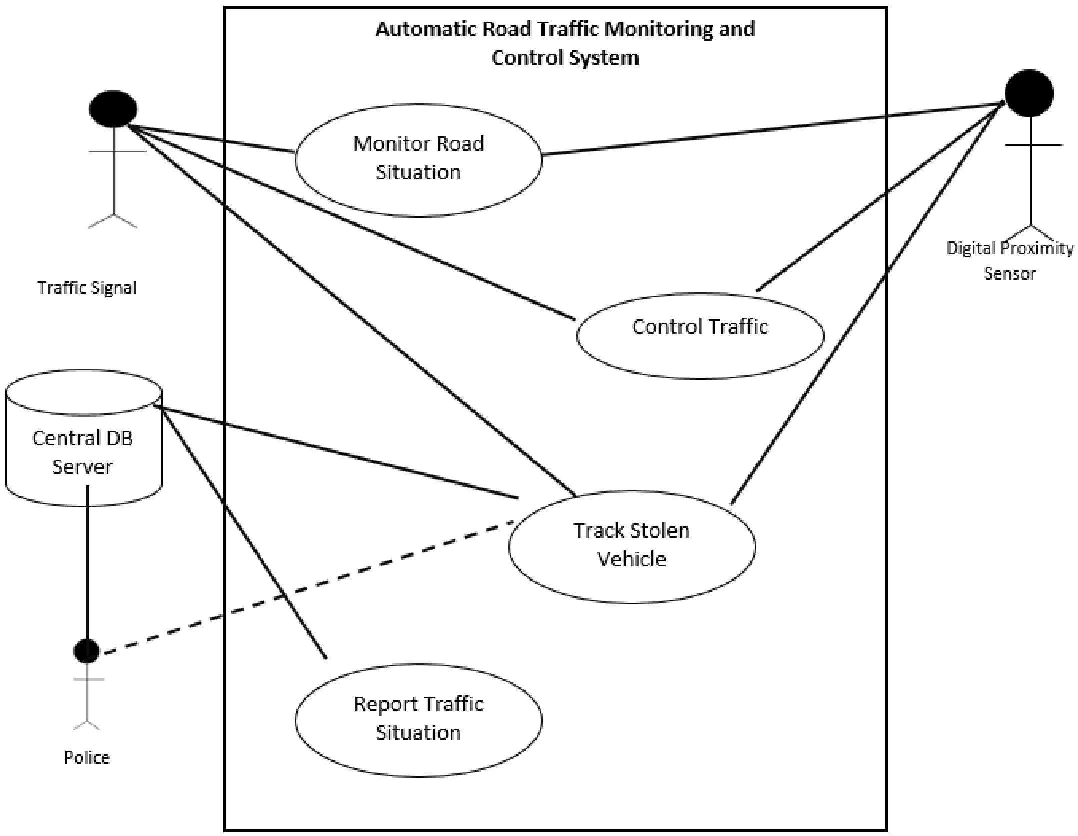

Figure 3.2: Use Case Diagram

The conceptual class diagram for the system as derived from Tables 3.1 to 3.4 is depicted in Figure 3.3.

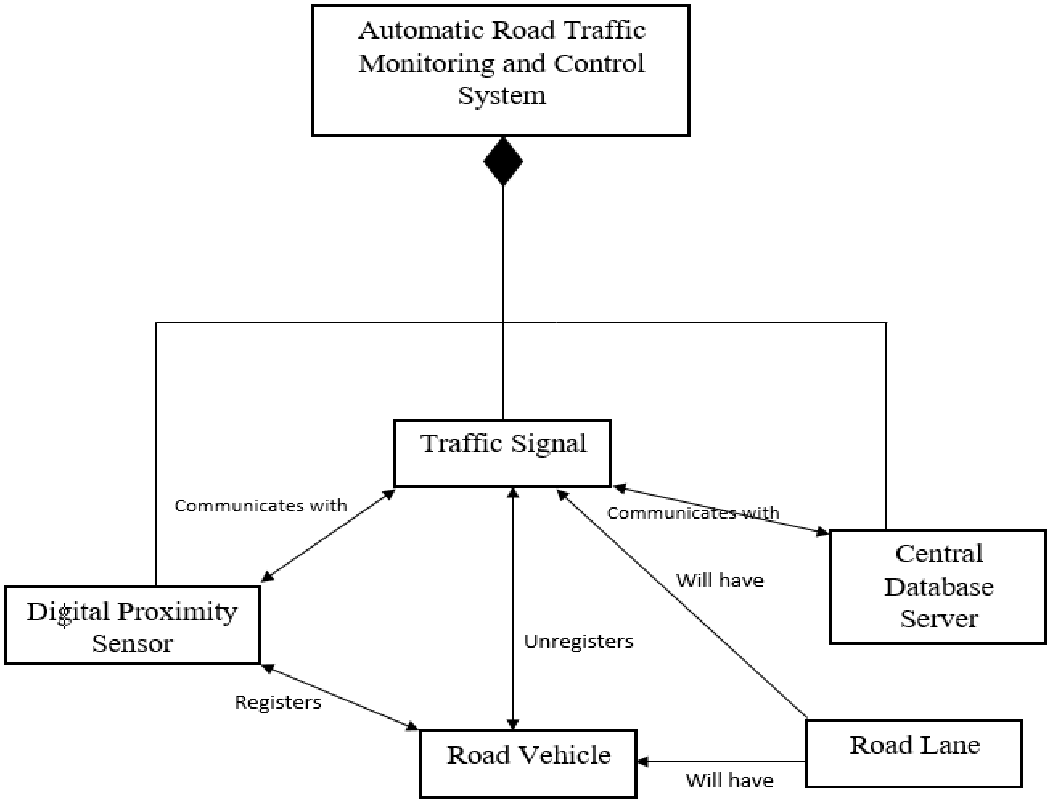

Figure 3.3: Conceptual class diagram of major modules of the system

From Figure 3.3, an automatic road traffic monitoring and control system is composed of the traffic signal lights, digital proximity sensor, and a centralised database server (cloud storage). A road vehicle which enters a traffic zone is registered by the proximity sensor and information sent to the traffic signal microcontroller which communicates the database. When the vehicle exits the traffic, zone the traffic signal captures this and unregisters the vehicle from the traffic. However, a road lane will have zero or more vehicles on it, and will also have one traffic signal observing the traffic situation for the specific lane.

From the conceptual class diagram, the block diagram representing the entire system is shown in Figure 3.4.

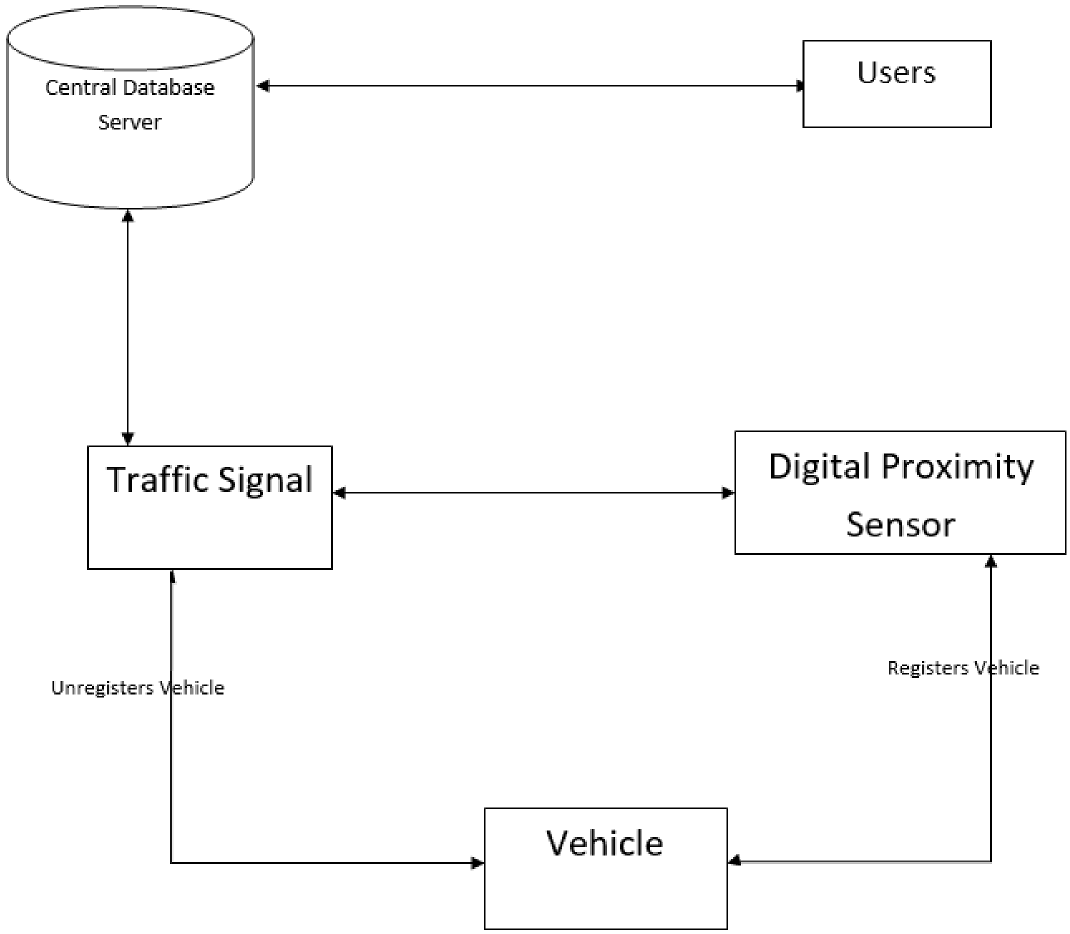

Figure 3.4: Block Diagram of Intelligent Traffic Control System

However, following the identified classes that make up the system, details of each class with respect to its data member and operations are represented in Figure 3.5.

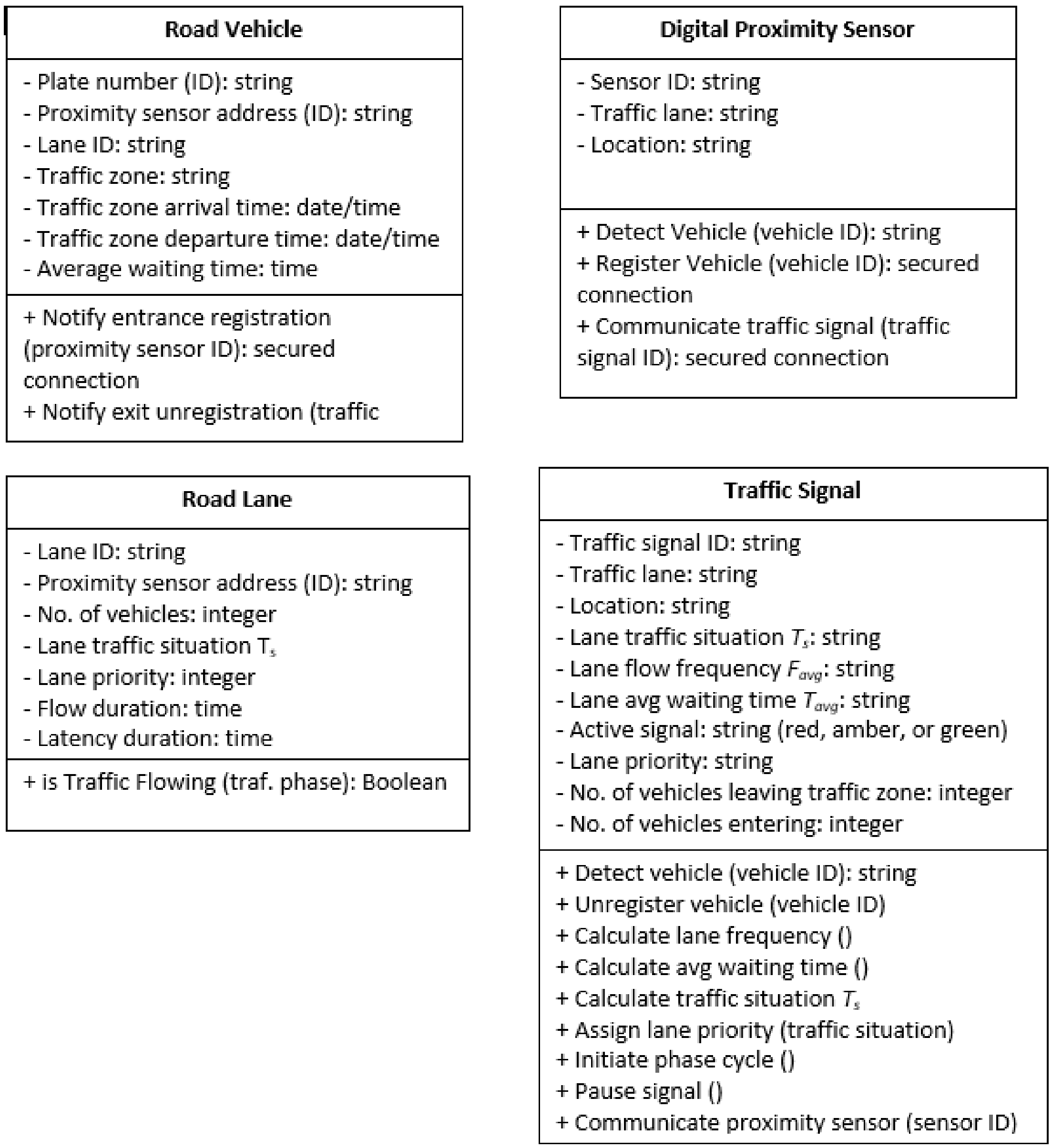

Figure 3.5: Identified entities, data objects, and operation of Intelligent Traffic Control System

3.2.2 Design of a Sensor Network

There are several technologies that can be applied for vehicular sensing. But considering minimal environmental impact and accuracy, this work deployed both inductive sensor array and PIR sensors for vehicular detection. These sensors are capable of detecting when a vehicle is present, not moving, and when a vehicle is moving. These sensors produce output voltages that have a relationship with the time domain. The voltages from these sensors are fed into a processing system that extracts the necessary information. The Figure 3.6 shows a block diagram representation of the vehicular sensing system.

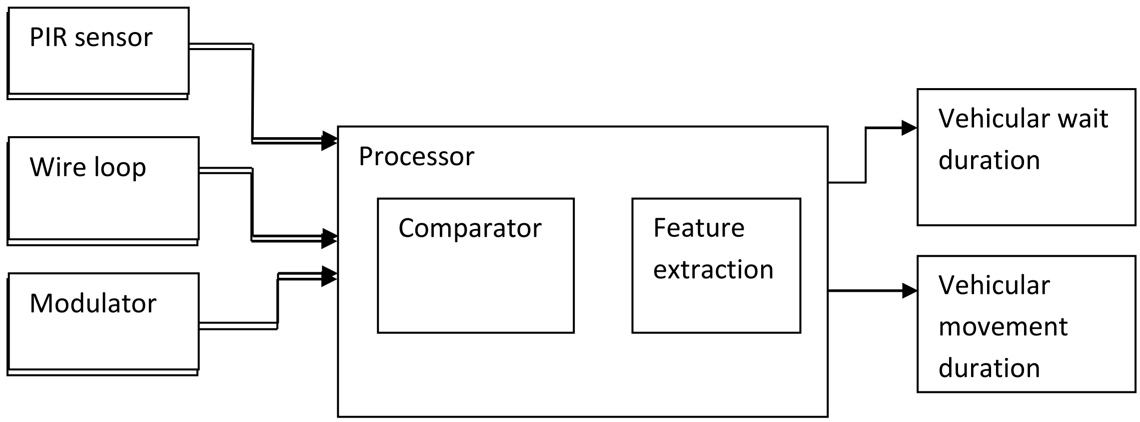

Figure 3.6: Vehicular sensing technology

3.2.3 Development of DTAT Algorithms Leveraging Fuzzy Logic and ANN

Given a junction that has four roads, the sensor array will report the traffic situation in each road. The vehicular wait duration and the vehicular movement duration as well as the length of the road and the speed limit of the road shall be used as input into the fuzzy logic system. These input parameters are fuzzified by making them pass through membership functions. The output of the membership functions goes through the fuzzy inference subsystem which is influenced by the fuzzy rule base system. The inference system produces an output membership function which is then defuzzified to get the output parameter values. The output parameter values are the optimized wait times and move times per road on the four-road junction.

3.2.3.1 DTAT Software Design

Following that traffic monitoring and control will require a distributed mechanism where different devices that make up the whole system shall be situated at separate locations, communicating with each other across a network in real-time, distributed computing paradigm such as load balancing and remote procedural call (RPC) are adopted for the implementation of the system.

Load balancing was used to strike a balance in the flow of traffic of different lanes of the road traffic, while RPC was used to remotely invoke/initiate operation(s) (procedure) of remote interconnected devices in order to facilitate real-time response of the embedded devices.

The operations of microcontrollers which drive the performance of the entire system are modelled and implemented as software. These operations include those of the Load Balancer, RPC, ANN, and Fuzzy Logic.

Different instances of the load balancer run at different lane traffic signal (the servers). The load balancer listens to incoming communication of different lane traffic signals (via RPC). This communication interchanges the *lane priorities* of the different traffic lanes to the different instances of the load balancer in order to update their respective priority records. The lane with the highest priority is assigned to initiate its phase cycle by it fuzzy controller, while other lanes pause. The green light duration (flow duration) of the traffic flowing lane depends on the priority level of the lane (which will depreciate with time), the priority level of other lanes.

The algorithm for the operations of the load balancer/RPC operations is represented in the flowchart in figure 3.6.

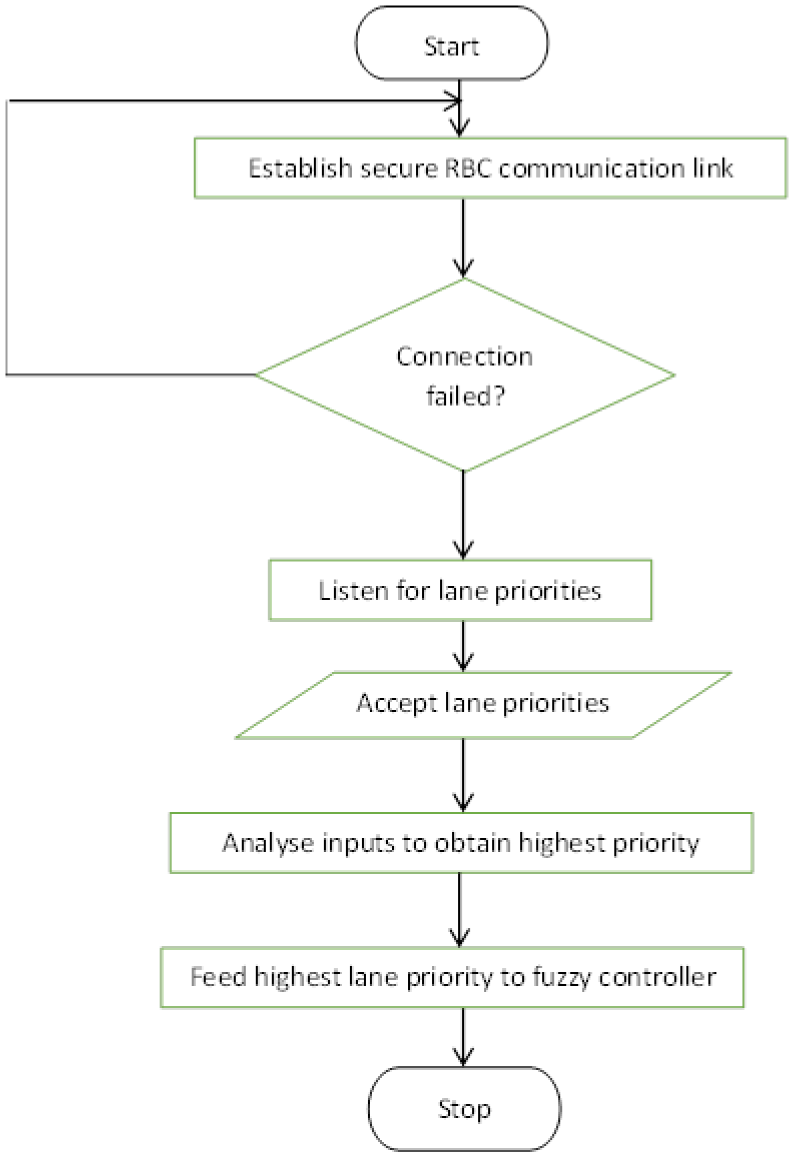

Figure 3.7: Flowchart for the operations of the Load balancer/RPC

3.2.3.2 Design of Fuzzy Controller

The fuzzy controller for a particular lane traffic signal accepts the highest lane priority from its load balancer, and performs its traffic signal control operation by processing the data. Hence, the operations of the fuzzy controller are modelled in the flowchart labelled Figure 3.7.

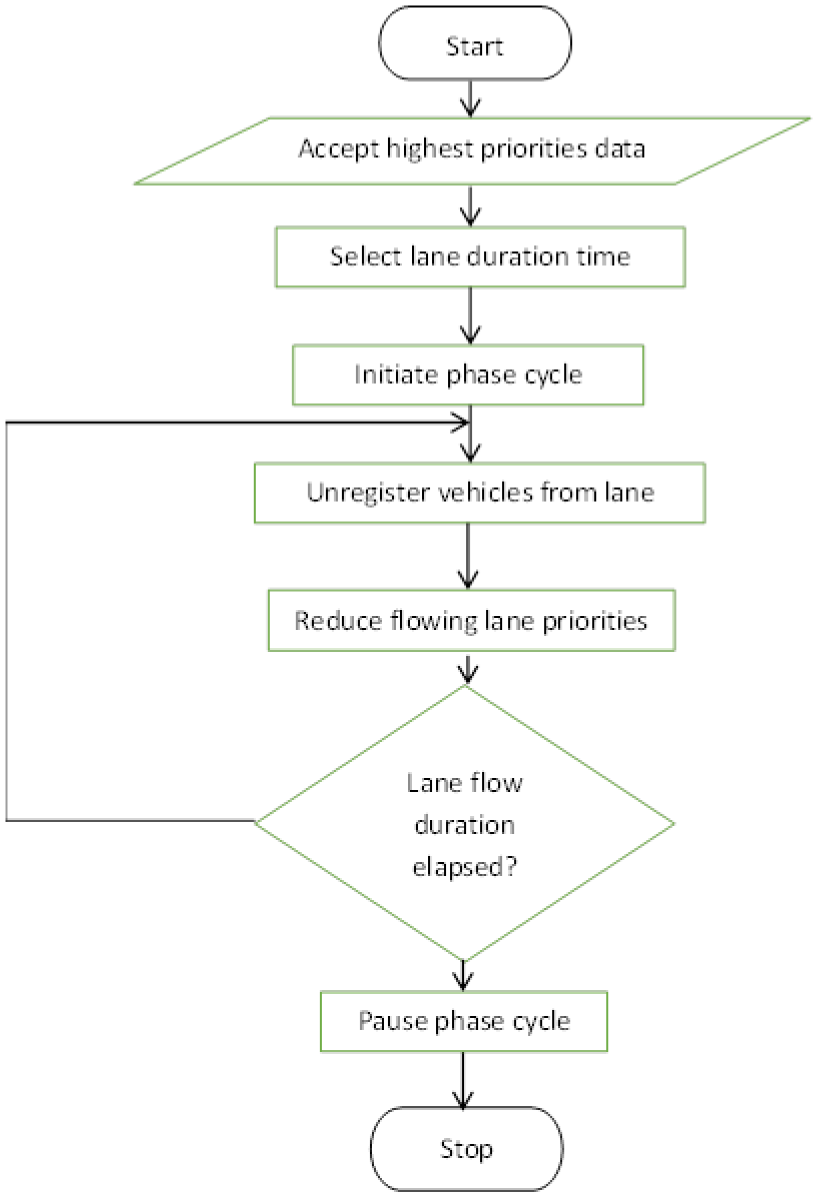

Figure 3.8: Flowchart for the operations of the Fuzzy Controller

Inference Rules for the Fuzzy Controller

In modelling the inference rules for the fuzzy controller which drives the logical flow of execution of operations of the system, it is worthy of note that the three stages of fuzzy logic implementation which involves fuzzification, defuzzification, and inference rules are all implemented by traffic controller system respectively by the sensors, actuators, and program.

In this case, the fuzzification process which involves the transformation of *crisp input*s to fuzzy inputs utilizes the following parameters – *wait time, go time, and vehicles on the lane.*

The inference rules for the fuzzy controller is similar to that of a human traffic controller reasoning process. With respect to the traffic control system, the following flow of execution occurs:

1. Allocate equal time to each lane of the road,
2. Approve go time for a specific lane and set light to green,
3. Start wait time count for other lanes,
4. Decrease go time for the approved lane,
5. Sense the amount of vehicles on the approved lane and wait time, relative to other lanes,
    a. If go time != 0 and vehicles very high,
        i. Maintain the green signal for lane.
        ii. Keep other lanes to on wait with red light.
        iii. Still count down the go time.
    b. If go time != 0 and vehicles high,
        i. Repeat steps 5a(i) to 5a(iii).
    c. If go time != 0 and vehicles normal,
        i. Check lane with highest wait time and highest vehicle.
        ii. Allocate go time to the lane and set light to green.
        iii. Repeat steps 3 – 5c(iii).
    d. If go time == 0 and vehicles normal,
        i. Repeat steps 5c(i) – iii.
    e. If go time == 0 and vehicles high,
        i. Increase the go time by 10% of the initially allocated go time,
        ii. Maintain the green signal,
        iii. Countdown the go time.
    f. If go time == 0 and vehicles very high,
        i. Repeat steps 5e(i) – iii.

From the steps highlighted above, it can be seen that the priority given to a specific lane determines whether traffic flows (green light) on that lane, or not (red light). This is further presented in the Table 3.5.

TABLE 3.5: FUZZY INFERENCE FOR LOW, MEDIUM OR HIGH PRIORITY LANE

Vehicles	**WAIT Time**	**GO Time**	**Lane Priority**	**RESULT**	**Traffic Signal**
HIGH	HIGH	= 0	HIGH	Allocate Go time to lane	Green
HIGH	= 0	> 0	HIGH	Increase Go time for lane	Green
NORMAL	HIGH	= 0	MEDIUM	Check for the lane with the highest wait time and vehicle.	Allocate Go time (Green) to lane with highest wait time and vehicle
NORMAL	= 0	> 0	LOW	Check lane with the highest wait time and vehicle	Allocate Go time to lane with the result.
LOW	HIGH	= 0	LOW	Check lane with the highest wait time and vehicle	Allocate Go time to lane with the result

With respect to this research, in order to enforce optimal control of the traffic signals for different road lanes, a firmware-based load balancer was implemented using a reversed sort of least-connection scheduling technique and made to run on each of the microcontroller attached to the traffic signal attached to every lane of the road which is situated at the road intersection.

The least connection scheduling technique which functions on the basis of assigning new client requests (in a network traffic) to the central microcontroller with the least number of current connections to clients was adopted in the load balancer to ascertain the traffic signal attached to the road lane with the worst case of traffic situation T_s i.e., the highest number of vehicles (clients). Thus, the number of vehicles in a traffic is ascertained by looking up the traffic queue data structure which stores the cars on the traffic queue for a particular lane in a specific period of time.

When the lane with the highest number of vehicles on traffic queue is detected, the green light for that lane is turned on while the red lights for the other lanes are turned on. If the load balancer detects equal traffic situation T_s for the four lanes it is monitoring, then the operations of a general scheduler which is based on Round Robin (RR) is invoked. This is however, rare to occur. Hence, the default operation of the traffic signals for the road lanes is based on RR where all traffic signal is allocated equal amount of time for the display of green light, and this display is done in a cyclic order as depicted in Figure 3.8 below.

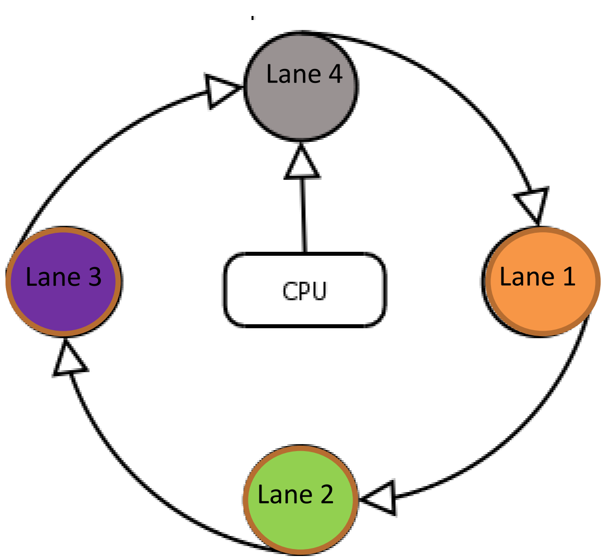

Figure 3.9: Round Robin execution of green light display

3.2.3.3 ANN Framework

Since the traffic monitoring and control system shall attempt to emulate the behaviour of a human traffic controller, it is necessary for it to have certain artificial intelligence (AI) features have embedded to the firmware. As a result of this, ANN and fuzzy logic are applied in the development of the software systems which drive the road traffic monitoring and control system.

The ANN framework would enable the system to monitor and learn traffic situations in order to make decisions of its own. This framework is however, imbibed in the firmware program that controls the operations of the traffic signal. In this case, some of the basic operations of the traffic signal become the functions of the nodes of the ANN. It is based on this reasoning that the ANN observes the flow of traffic data, learns from it, and makes its decision(s) which invokes the fuzzy logic.

With respect to the operations of the nodes, it is important to consider the dynamics of activities associated with the traffic signal as regards the input data, basic operations, and output results. Thus, the interaction between the nodes is represented in Figure 3.10.

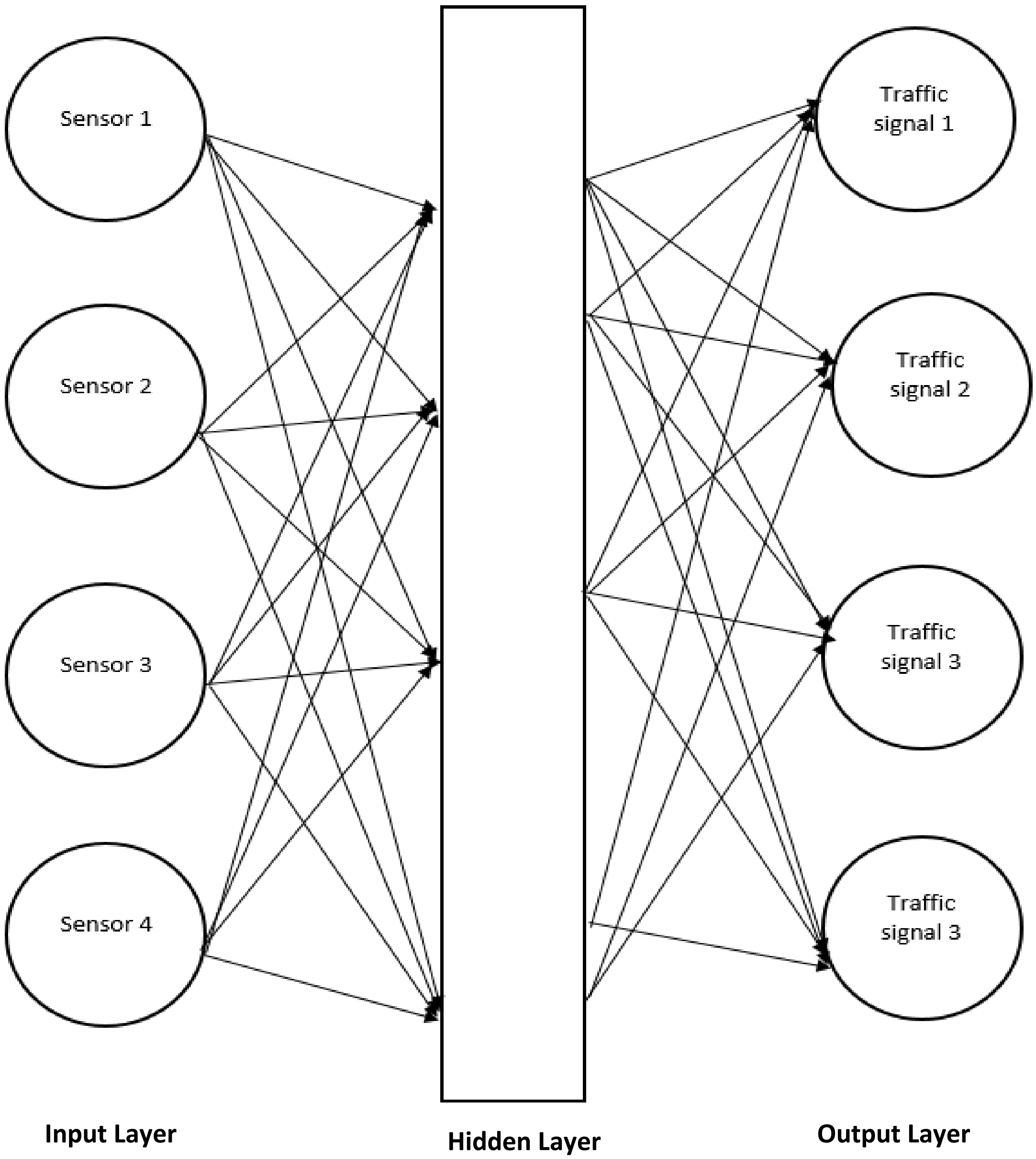

Figure 3.10: ANN Framework of the System

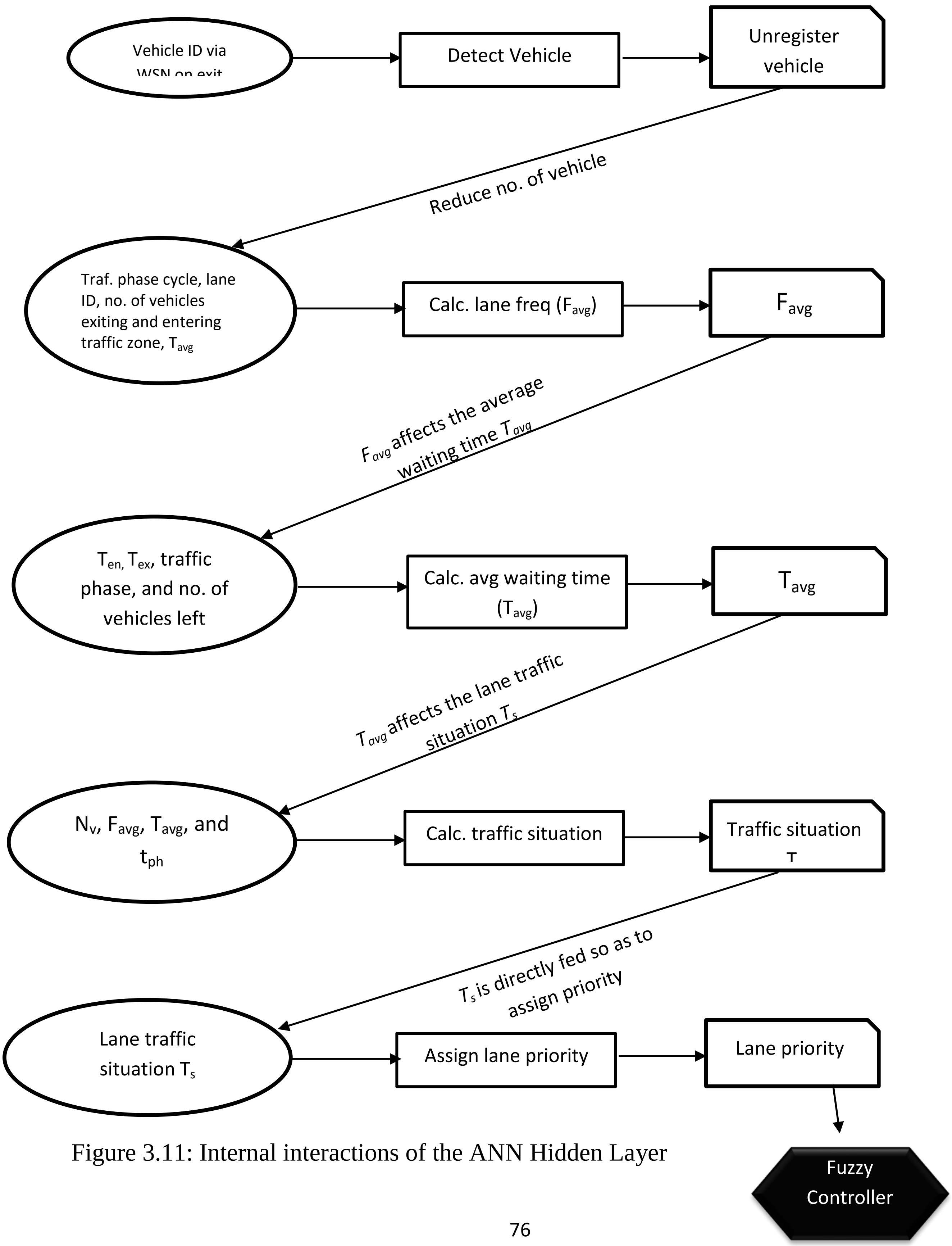

Figure 3.11: Internal interactions of the ANN Hidden Layer

From Figure 3.11 above, it could be seen that the interactions among the activities of the hidden layer influences one another in somewhat. When a vehicle is unregistered for a particular lane, it directly influences the number of vehicles used in calculating the lane frequency F_{avg}. Same applies to other nodes and down to the ASSIGN LANE PRIORITY operation which calculates lane priority and feeds its output to the fuzzy logic component in order to control the lane traffic phase cycle.

At this point, it is pertinent to note the separation of functions between the ANN and the Fuzzy logic controller in that the former observes the traffic situation and does monitoring based on set of captured data, while the latter performs the function of control via the fed result of the monitored lane traffic situation. However, the operations of the fuzzy logic controller are considered as follows;

1. Initiate phase cycle: this takes in the lane priority as input in order to ascertain which lane to initiate cycle. The output result is to change the traffic signal of the road lane with the highest priority from red, through amber to green. The signal remains at green for a specific duration of time based on the assigned priority with respect to those of other lanes. This duration of time is known as the "flow duration" as captured in the Road Lane entity in Figure 3.6.

2. Pause phase cycle: this operation also functions based on the priority of the road lanes. The output result is to change the traffic signal of the road lane with the least priority from green, through amber to red. The signal remains at red for a specific duration of time based on the assigned priority with respect to those of other lanes. This is known as the "latency duration" as depicted in the Road Lane entity in Figure 3.6.

3. Register vehicle: this is the operation of the traffic light (running the fuzzy controller) that records vehicles that come into a particular road lane (via communication with the motion proximity sensor) in order to increment the number of vehicles that are within the traffic zone of the lane. This enables the

ANN to carry out some operations as described in figure 3.8 as to determine the traffic situation T_s.

3.2.4 Development of Camera System and Traffic Offence SMS Gateway

This sub-section discusses the development of camera system and SMS gateway for capturing traffic offenders. The circuit diagram for implementing the camera system and SMS gateway is shown in Figure 3.12.

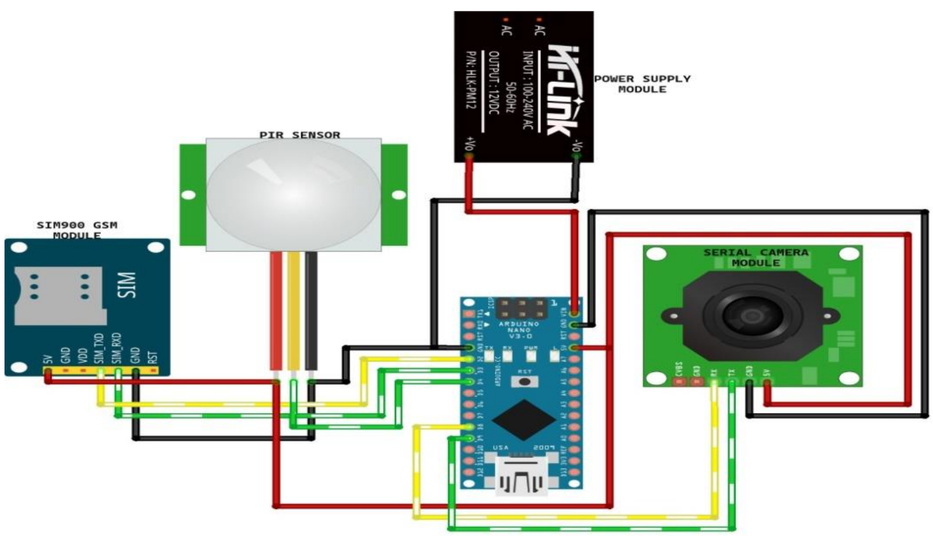

Figure 3.12: Circuit diagram for implementing the camera system and SMS gateway

3.2.4.1 Camera System

When a driver beats the traffic light, the traffic management system will use one of its sensors to detect it and immediately activate the camera which will start taking snap shots of the vehicle license plate number. The images so acquired by the camera will then go through an image recognition algorithm in other to extract the license plate number.

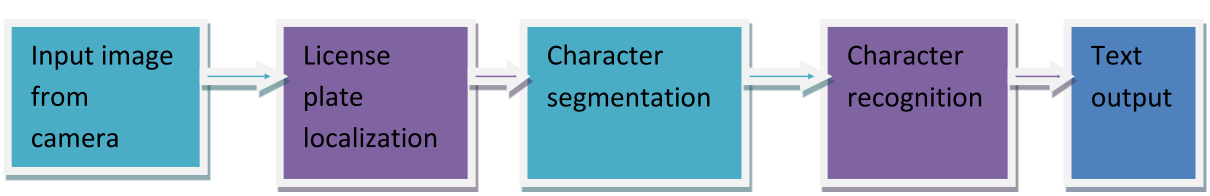

Figure 3.13: Image recognition block diagram for license plate number extraction

With the license plate number gotten, it will be used as a primary key in the FRSC database in other to retrieve from the database the phone number and the address of the vehicle owner. With the information (phone number) the SMS gateway will then be used to send an SMS to the vehicle owner; telling him his offence, details of the offence and fine to be paid.

3.2.4.2 SMS Gateway

The GSM module is a portable hardware device that can be configured as an SMS gateway. There are many GSM modules, but SIM900A, manufactured by SIMCON, has found wide scale application being used as SMS gateway.

AT command is the communication syntax used by devices to communicate with the GSM module in order to set up an SMS gateway. This AT command specifies communication parameters like baud rate, echo, network registration, SMS indication and network connected status. Figure 3.14 depicts the flowchart to set up an SMS gateway

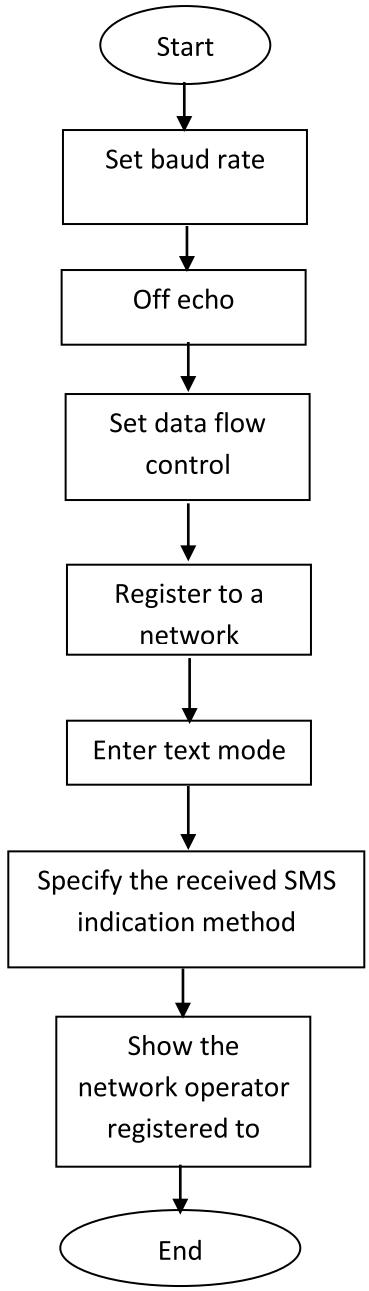

Figure 3.14: How to set up SMS gateway

3.2.5 System Simulation and Testing

Simulations of the system as well as tests carried out on the system are discussed in this sub-section.

3.2.5.1 System Simulation

The system is simulated using Proteus 8 Professional on a Windows OS PC running on a RAM memory of 8GB and HDD of 1TB with a dedicated graphics memory of 4GB. The simulation was implemented on the basis of Dynamic Time Allocation Technology (DTAT) in combination with fuzzy logic running on an ANN. It is a four-way traffic controller system where at every given point when the traffic signal is triggered, a vehicle in traffic can take three possible directions. In so doing, the following components were used:

i. Microcontroller – which executes the assembly level program that implements the DTAT, fuzzy logic, and ANN. ANN, which is based on a collection of interconnected units that are loosely linked to the brain (microcontroller in this case) is typically demonstrated as the respective traffic signals attached on each road lane is connected to the microcontroller. Fuzzy logic and DTAT are the basis upon which the traffic lights are controlled as coded in the program which executes in the microcontroller.

ii. Connectors – these interconnect the components together and conversely establish the ANN in the system.

iii. Traffic lights – used to physically transmit the signals on each lane of the roads.

iv. LED timers – attached to each traffic light in order to display signal time allocations to the vehicles in traffic.

v. Interconnecting wires – used for interconnecting the components.

The system was then simulated in two different versions. The first version indicated the use of Arduino uno microcontroller for collecting data for each of four road lanes at the physical intersection, and the use of interconnectors for neat presentation. The second indicated the use of Arduino nano microcontroller for coordinating the camera and SMS gateway system as well as the use of wires for interconnection and LED display timers for indication of signal time to the vehicles. The overall simulated circuit is shown in figure 3.15.

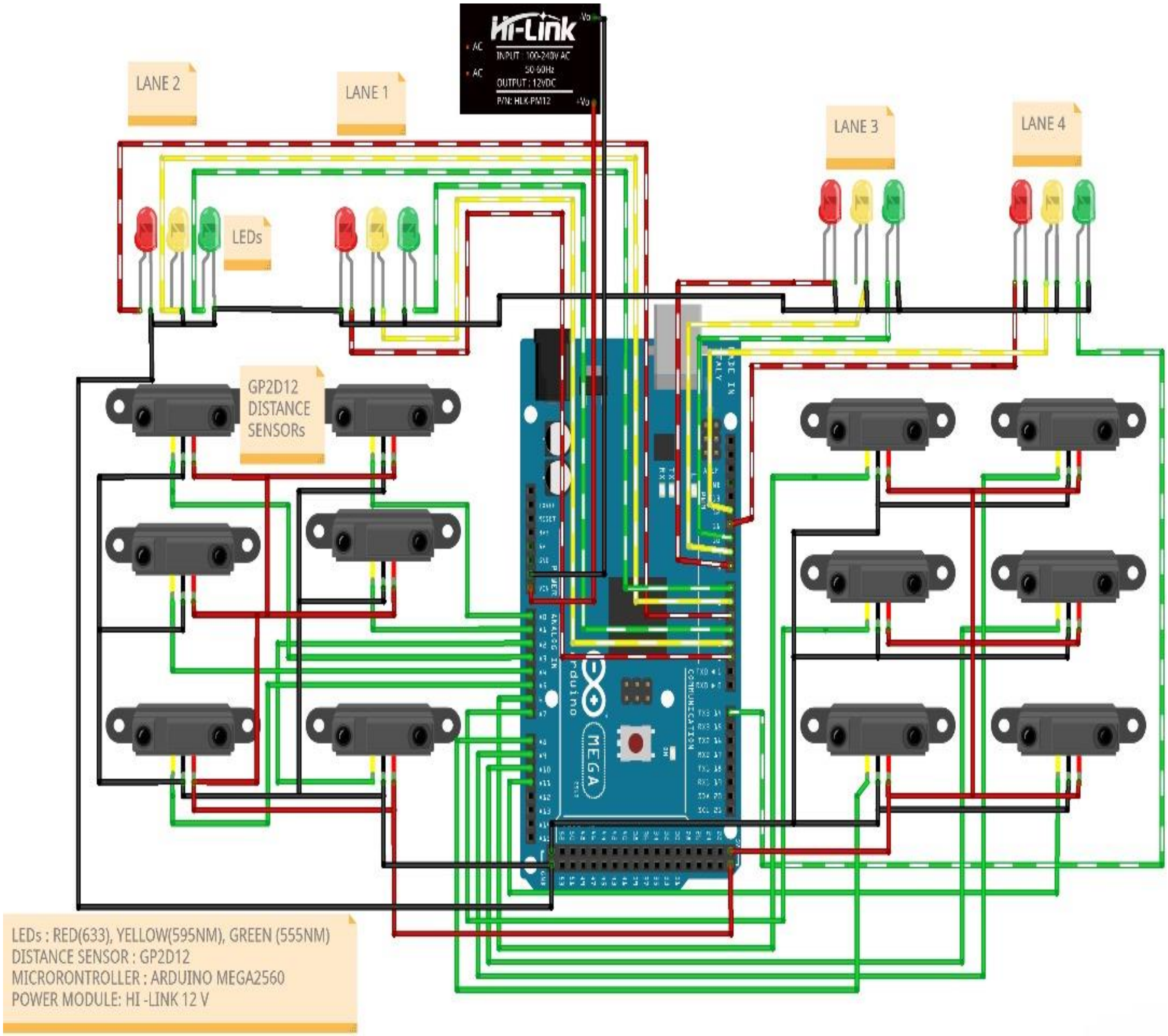

Figure 3.15: Simulated Circuit Diagram

Flowchart of the System

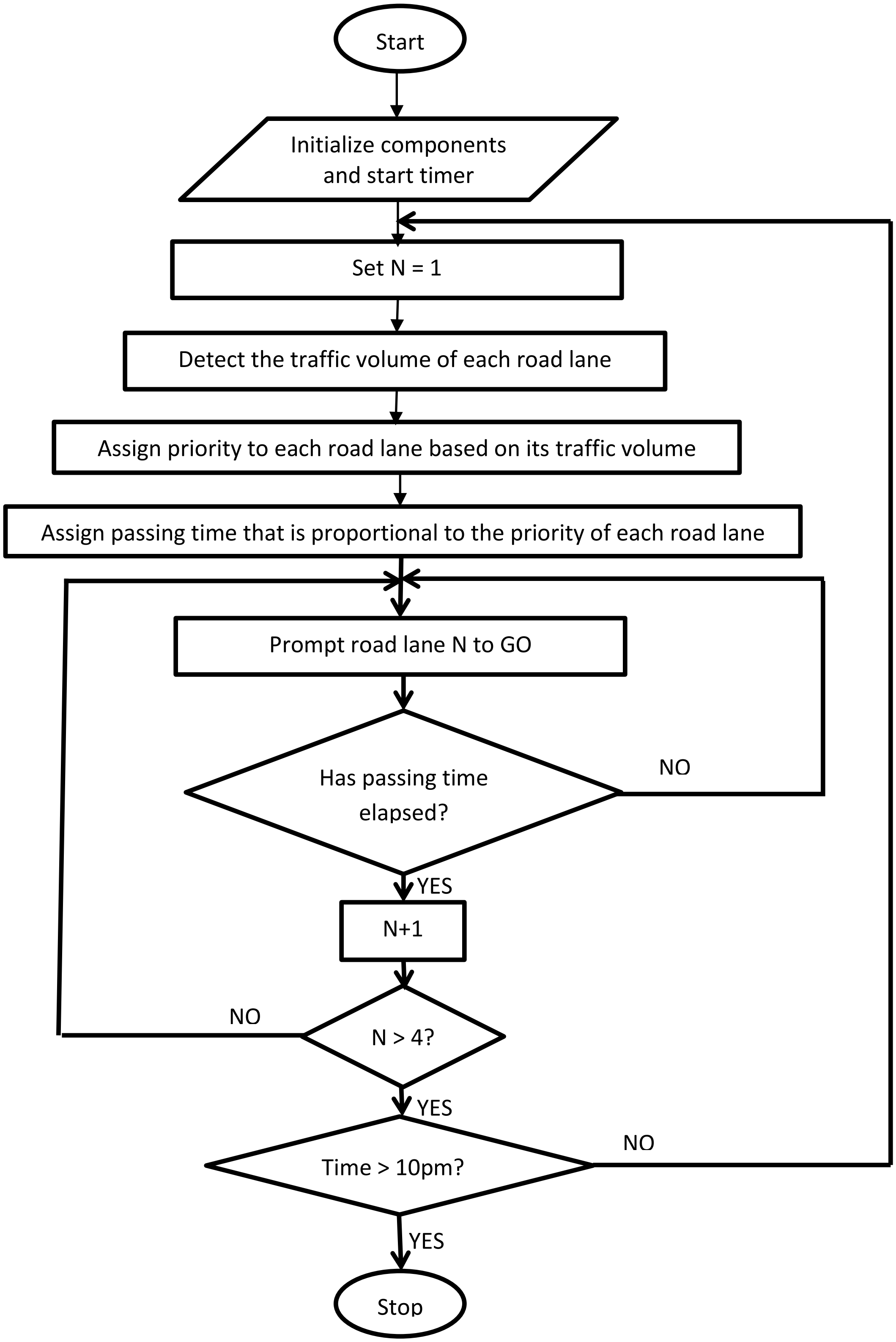

Figure 3.16: Intelligent Traffic Management System Flowchart

It is imperative to note that the flowchart in Figure 3.16 is a continuous loop that would only pause at 10:00PM when it is assumed that the traffic is minimal. This is due to the fact that the simulated traffic signal control system is a real-time system. Hence, based on the distributed time allocation algorithm, the system continues to run in real-time till 10:00PM.

3.2.5.2 Testing the System

The intelligent traffic management system was tested in a superficially created road intersection with up to four lanes. The Arduino Uno microcontroller which contains the traffic signal LEDs was stationed at the intersection while the proximity sensors were positioned at intervals up to 400 metres away from the intersection for each lane. Wooden blocks which represent road vehicles moved into and out of the lanes (traffic zone) following the change of signal of the LEDs attached to the microcontroller for each lane. Therefore, the proximity sensor which is attached to each lane senses when vehicles enter the lane, and the microcontroller at the intersection senses the departure of vehicles from the lane. It is based on this that the traffic signals at the intersection are coordinated and controlled via the microcontroller.

CHAPTER FOUR
RESULTS AND DISCUSSIONS

4.1 Results

This section presents the results obtained from modeling, designing, simulating and testing the system. Following that the traffic situation Ts is affected by certain factors such as the number of vehicles Nv, average waiting time of vehicles in traffic zone of road lanes Tavg, and average frequency at which vehicles enter and leave traffic zone Favg; these factors are considered the key parameter indices (KPIs) upon which the results were obtained and discussed. These KPIs were tested for the system.

4.1.1 UML Design Pattern

The result obtained by utilizing UML design pattern in modelling the intelligent traffic management system is presented in Figure 4.1.

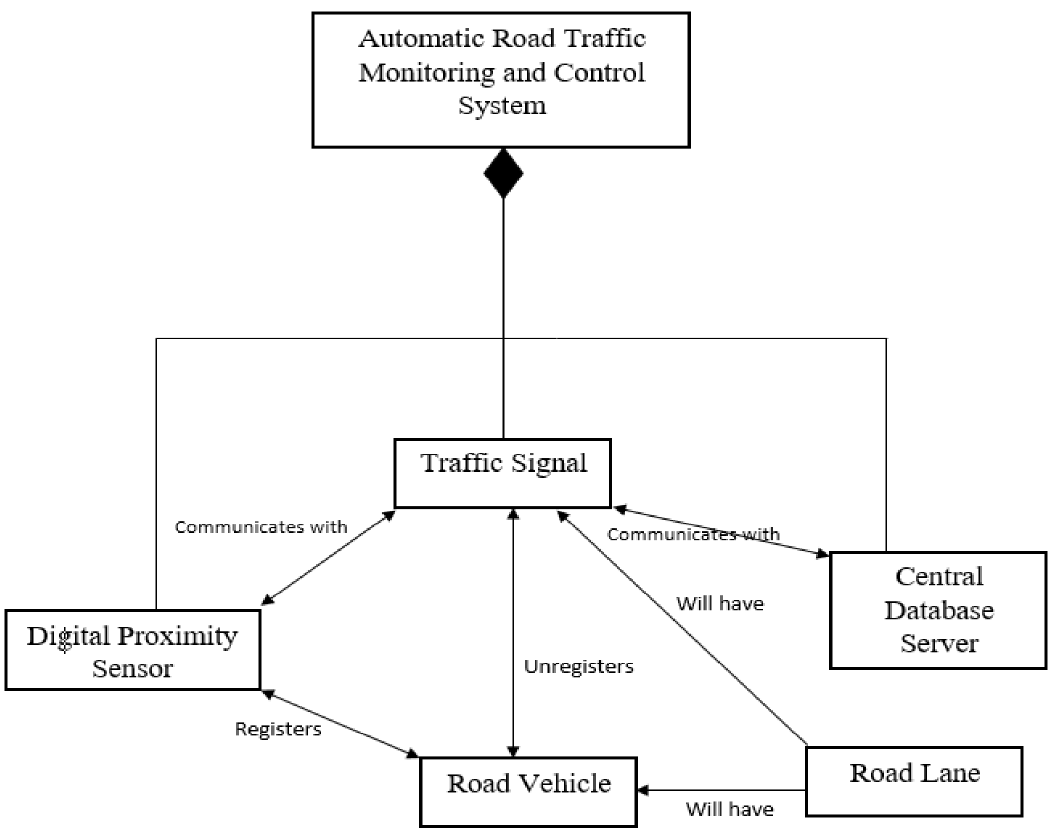

Figure 4.1: A model of the developed system

4.1.2 Sensor Network

Apart from the number of vehicles N_v, the parameters T_{avg}, F_{avg}, and t_{ph} are all dependent on time. Having considered that the test was carried out in an isolated cross-road where four traffic signals were used, the number of vehicles that were made to enter the traffic zone N_e and those that left the zone N_l within a specific time interval representing the traffic phase cycle t_{ph} varied for different lanes. However, the traffic zone for each of the lanes remained the same. For each lane, this is the distance between the proximity sensor and the point at which the lanes intersect. The proximity sensor is stationed 400 metres away from the intersection, and this represents the traffic zone distance. On the average, the traffic zone can accommodate a total number of 100 cars.

The traffic zone is divided into two equal subzones. These are;

1. The high congestion zone, and
2. The low congestion zone.

The traffic zone was subdivided in order to facilitate optimal performance of the system as lanes with vehicles on-queue up to the high congestion zone are issued higher priority than those without vehicle on-queue up to the high congestion zone. The low congestion zone is 0 metre – 200 metres from the lane intersection while the high congestion zone is 200 metres – 400 metres away from the intersection.

At every traffic phase cycle t_{ph}, two lanes (L1 and $L1^{|}$) are freed to flow while the flow of traffic in the other two lanes (L2 and $L2^{|}$) is halted as represented in Figure 4.2.

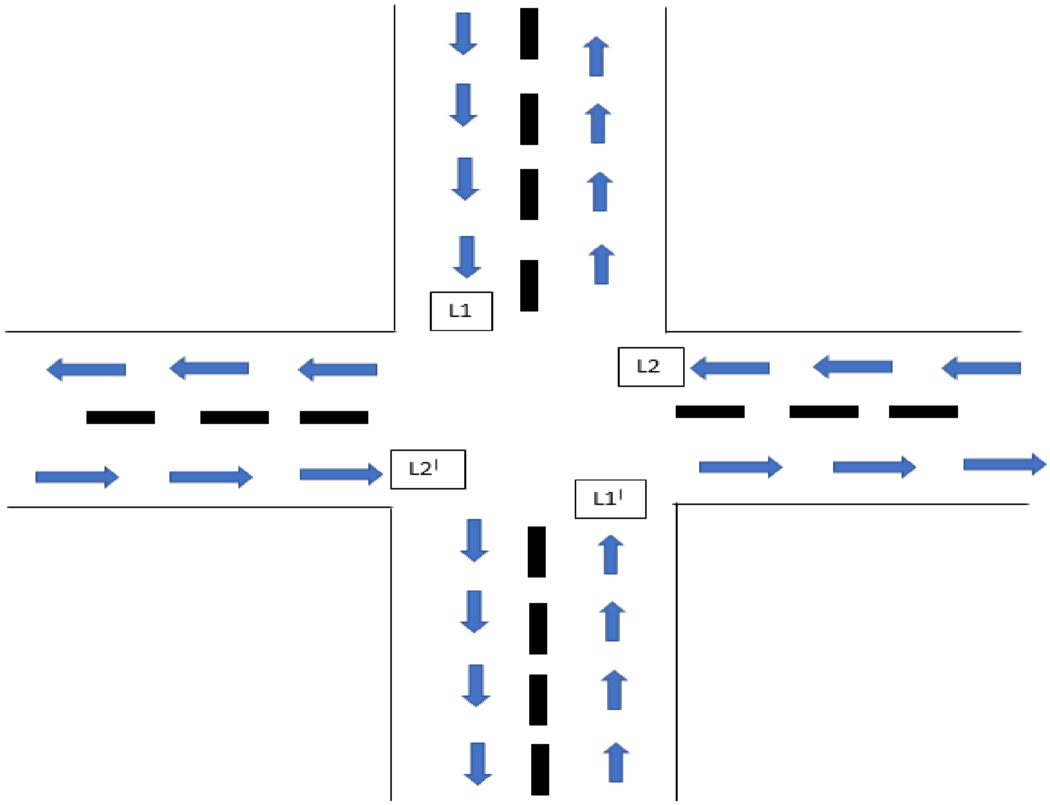

Figure 4.2: Lane intersection showing flow mechanism

4.1.3 DTAT Algorithms Leveraging Fuzzy Logic and ANN

Four signal generators were used to feed input signals into the microcontroller which runs the ANN and fuzzy logic program. The microcontroller then processes the input signals and produces outputs. Each input signal represents the traffic density of each lane. Thus, the four input signals represent the four lanes of the road making the intersection. Priority issued (from a scale of 1 – 10) to each lane (input signal) is based on the level of signal for the lane in relation to other lanes. It is based on the lane priority that the flow duration and wait duration is assigned to each lane relative to other lanes. Therefore the traffic situation Ts is a function of the level of input signal, flow duration, and wait duration of each lane of the road. Tables 4.1 – 4.4 show the test results for the four input signals.

Table 4.1: Relationship Between Lane Priority, Flow Duration, and Wait Duration for the First Traffic Phase $T_{ph1.}$

Lane	**Traffic Density**	**Lane priority**	**Flow duration (s)**	**Wait duration (s)**
1.	150	7.50	100	215
2.	200	10.00	120	195
3.	87	4.35	55	260
4.	48	2.40	40	275

Table 4.2: Relationship Between Lane Priority, Flow Duration, and Wait Duration for the Second Traffic Phase $T_{ph2.}$

Lane	**Traffic Density**	**Lane priority**	**Flow duration (s)**	**Wait duration (s)**
1.	72	4.50	60	180
2.	33	2.00	20	220
3.	180	10.00	120	120
4.	55	3.40	40	200

Table 4.3: Relationship Between Lane Priority, Flow Duration, and Wait Duration for the Third Traffic Phase $T_{ph3.}$

Lane	**Traffic Density**	**Lane priority**	**Flow duration (s)**	**Wait duration (s)**
1.	100	5.00	60	185
2.	200	10.00	120	125
3.	70	3.50	45	200
4.	40	2.00	20	225

Table 4.4: Relationship Between Lane Priority, Flow Duration, and Wait Duration for the Fourth Traffic Phase T_{ph4}.

Lane	Traffic Density	Lane priority	Flow duration (s)	Wait duration (s)
1.	87	4.50	60	180
2.	48	2.24	40	200
3.	33	2.00	20	220
4.	180	10.00	120	120

4.1.4 Camera System and SMS Gateway for Traffic Offence

The result obtained from camera system and SMS gateway is shown in Table 4.5.

Table 4.5: Results Obtained from Camera System and SMS Gateway

S/N	Vehicle Plate Number	Date	Time	Offence	Fine	Driver's Phone Number
1	APP62CA	02/04/2024	5:15PM	Driving Against Traffic	N50,000	08065128423
2	RUM739FK	02/04/2024	5:21PM	Beat Traffic Light	N20,000	09087642334
3	PHC471MB	02/04/2024	5:25PM	Driving Against Traffic	N50,000	08162387641
4	ABJ362PT	02/04/2024	5:30PM	Driving Against Traffic	N50,000	09054286541

4.1.5: Simulation Results

In the Proteus Simulation environment, the four signal generators were used to feed input signals into the microcontroller which runs the ANN and fuzzy logic program. The microcontroller then processes the input signals and produces outputs which are displayed in the oscilloscope. The microcontroller accepts 0V – 0.8V as LOW, and 2V – 5V as HIGH. The number of vehicles leaving a lane, N_l and the number of vehicles entering a lane, N_e were varied for different lanes within specific traffic phase t_{ph}. Thus, t_{ph} is made constant as the cycles must continue with respect to time for the different lanes. Albeit the t_{ph} is kept constant, it is in a way affected by the green time duration which is a function of lane priority. Hence, this poses some execution constraints as it leads to variations of t_{ph}. This is addressed by the fuzzy logic which handles the real-time control of the traffic signal.

The traffic situation T_s is actually represented by different priority levels. The traffic situation T_s can be;

1. Free,
2. Normal,
3. Medium, and
4. Congested

Conversely, priority levels are set to 1 (free condition), 2 (normal condition), 3 (medium), and 4 (congested condition). Results for variation in number of vehicles for different lanes with respect to four consecutive traffic phases $t_{ph1} - t_{ph4}$. The signal detections for different road lanes are shown in plates 4.1 - 4.4 below.

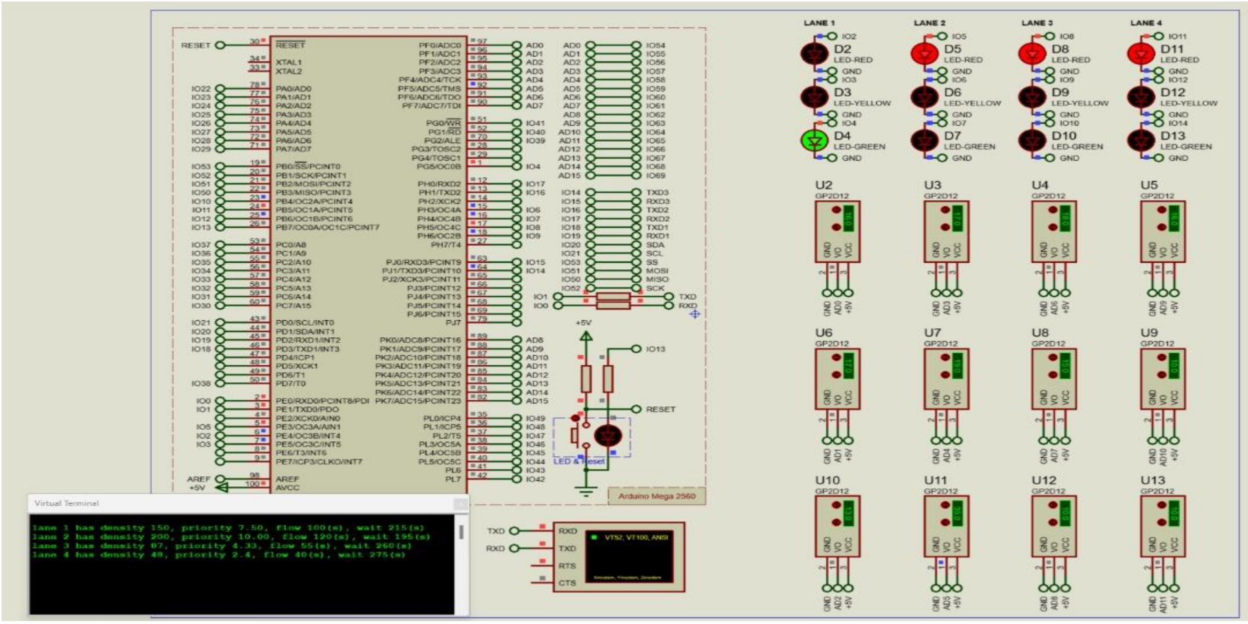

Plate 4.1: Simulation Image for lane 1

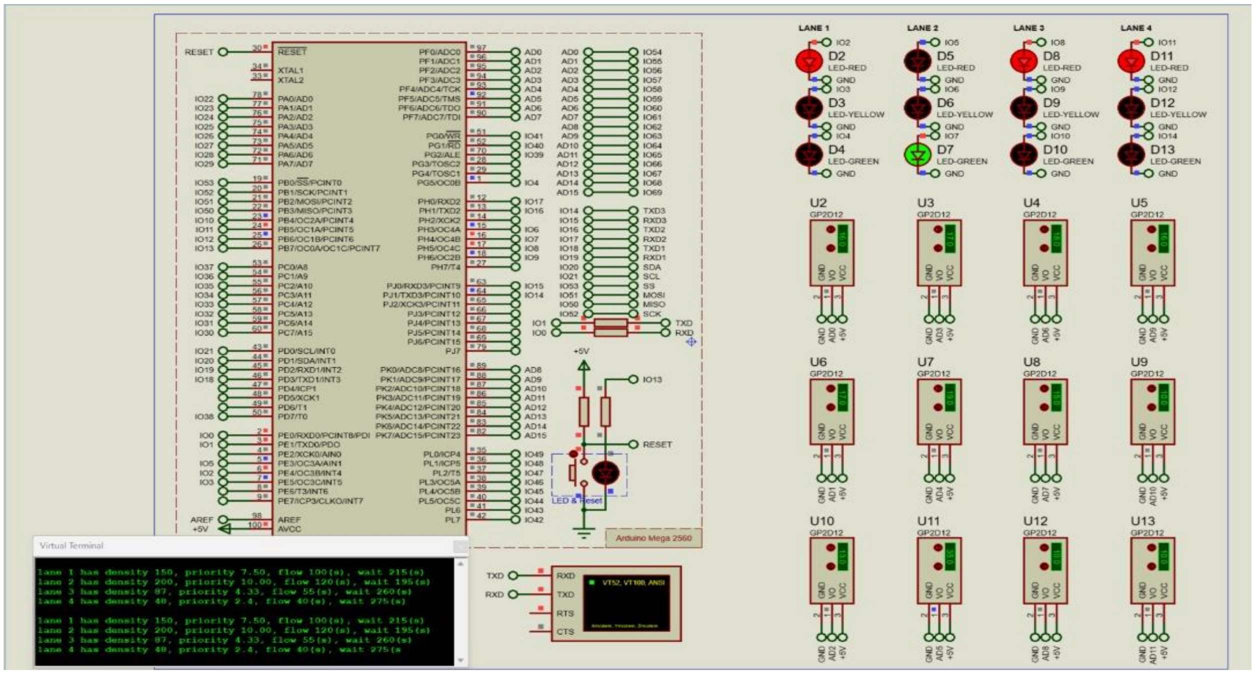

Plate 4.2 Simulation Image for lane 2

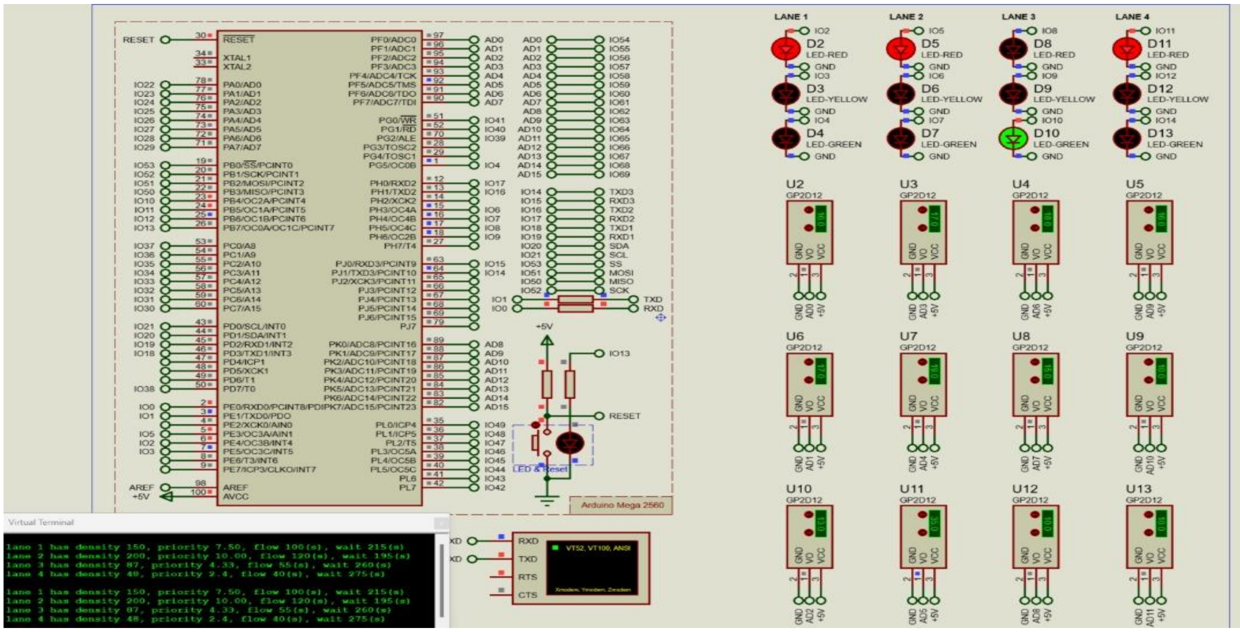

Plate 4.3: Simulation Image for lane 3

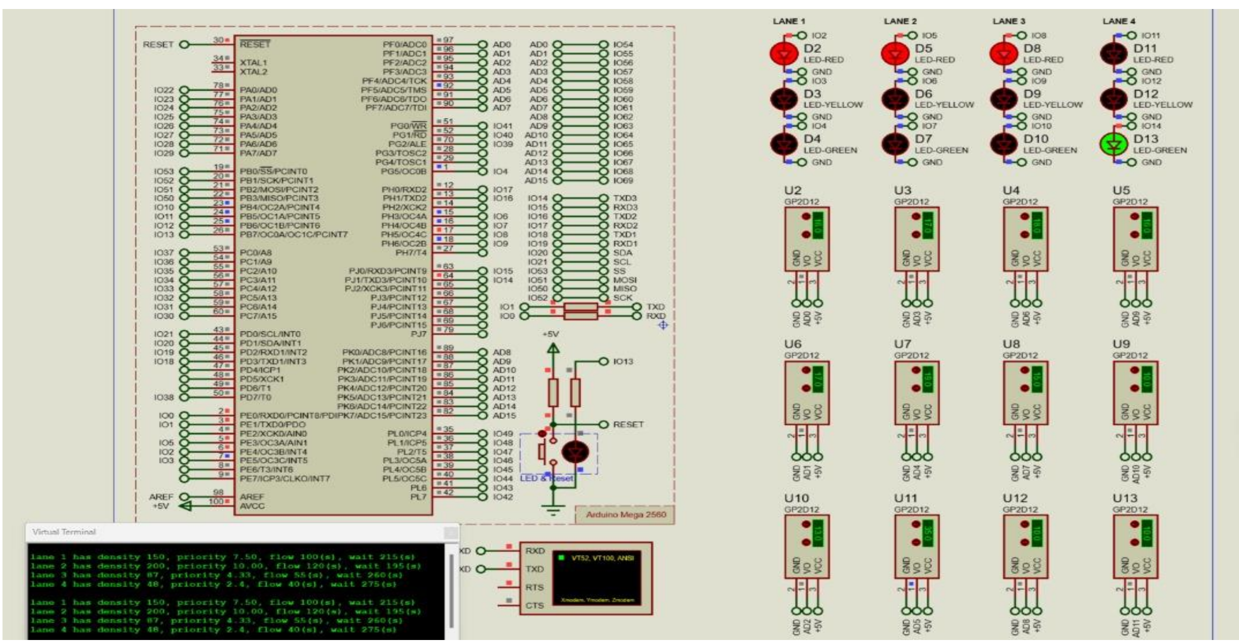

Plate 4.4: Simulation Image for lane 4

Table 4.6: Relationship between Variation of Number of Vehicles Entering and Leaving Different Traffic Lanes and Traffic Condition for Traffic Phase T_{ph1}

Lane	**N_l**	**N_e**	**T_{avg} (s)**	**F_{avg}**	**Lane priority**	**Flow duration (s)**	**Wait duration (s)**
1.	80	60	300	0.003	8.00	80	190
2.	100	10	180	0.005	10.00	100	170
3.	50	5	500	0.002	5.00	50	220
4.	40	45	350	0.003	4.00	40	230

Table 4.7: Relationship between Variation of Number of Vehicles Entering and Leaving Different Traffic Lanes and Traffic Condition for Traffic Phase T_{ph2}

Lane	**N_l**	**N_e**	**T_{avg} (s)**	**F_{avg}**	**Lane priority**	**Flow duration (s)**	**Wait duration (s)**
1.	30	60	400	0.003	2	60	330
2.	40	10	250	0.004	3	90	300
3.	80	5	60	0.017	5	150	240
4.	40	45	56	0.018	3	90	300

Table 4.8: Relationship between Variation of Number of Vehicles Entering and Leaving Different Traffic Lanes and Traffic Condition for Traffic Phase T_{ph3}

Lane	**N_l**	**N_e**	**T_{avg} (s)**	**F_{avg}**	**Lane priority**	**Flow duration (s)**	**Wait duration (s)**
1.	30	60	200	0.005	2.5	50	140
2.	40	10	120	0.008	3.0	60	130
3.	40	20	480	0.002	3.0	60	130
4.	10	30	350	0.003	1.0	20	170

Table 4.9: Relationship between Variation of Number of Vehicles Entering and Leaving Different Traffic Lanes and Traffic Condition for Traffic Phase T_{ph4}

Lane	**N_l**	**N_e**	**T_{avg} (s)**	**F_{avg}**	**Lane priority**	**Flow duration (s)**	**Wait duration (s)**
1.	20	40	300	0.003	2.0	40	140
2.	40	10	220	0.005	3.0	60	120
3.	40	20	212	0.005	3.0	60	120
4.	10	30	150	0.007	1.0	20	160

4.2 Discussions

Relating the results presented in Tables 4.1 – 4.4, it is obvious that the signal level, which represented the traffic density influenced the priority assigned to each lane. Thus, the higher the signal level, the higher the priority and consequently, the higher the assigned flow duration, and the lower the wait duration (Gedik, 2020). This implies that in the actual sense, high traffic density for a specific lane will automatically assign the lane with a higher priority which will result to increase in the flow of traffic for the lane, thereby resulting to even distribution in the traffic flow for the various lane. By this reasoning, there is relatively an even distribution of the flow duration (as well as the waiting time) of the lanes for the four traffic phases $T_{ph1} - T_{ph}$ (Deulkar, 2019).

Results in Tables 4.6 – 4.9 show that for a specific phase cycle, two sets of the lane (ln and $ln^{|}$, where n is the lane ID) are made to flow for a specific period of time (flow duration) which runs in seconds. This is based on the priority level of the lane which is predominantly influenced by the average waiting time of vehicles on the traffic queue, and the average frequency at which vehicles flow in each flow duration (Chen & Liu, 2021). It could be observed that as a specific set of lane is made to flow, the priority for those set of lanes reduces, the average frequency of flow of vehicles on the lanes increases thereby reducing the congestion, and consequently the average waiting duration of vehicles decreases. Table 4.10 presents the interrelationship between the different lanes, average frequency of flow, and waiting time of vehicles.

Table 4.10: Relationship Between Average Frequencies for Different Lanes at Different Traffic Phases t_{ph}

Lane	T_{ph1} F_{avg}	T_{ph2} F_{avg}	T_{ph3} F_{avg}	T_{ph4} F_{avg}
L1	0.003	0.003	0.005	0.003
L1$^{|}$	0.005	0.004	0.008	0.005
L2	0.002	0.017	0.002	0.005
L2$^{|}$	0.003	0.018	0.003	0.007

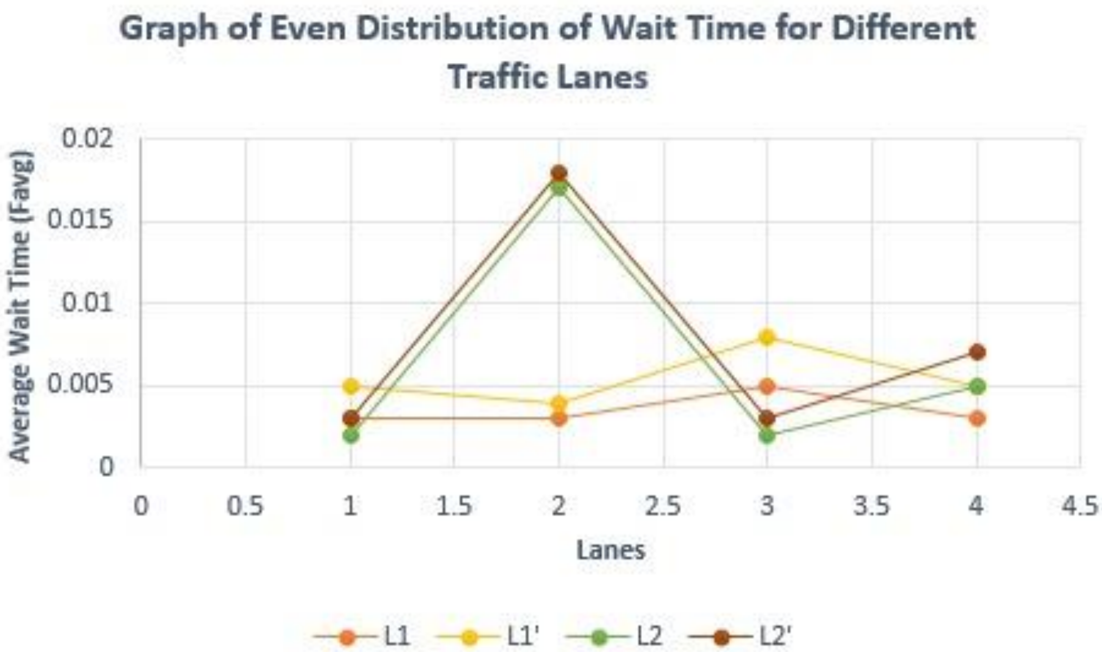

Figure 4.3: Even distribution of frequencies for different lanes on various traffic phase.

Figure 4.3 technically shows the even distribution of average frequencies F_{avg} among the four lanes (L1 – L1`) for four different traffic phases. From the diagram, it is observed that though there was a sharp increase in the frequency of the third lane L2 at the second phase of t_{ph2}, it could be seen that there was a relative drop in the frequency in the next phase. This is relative to other lanes (Rida et al, 2018). Hence, if the results are extrapolated, F_{avg} for the various lanes will all converge at a point. This clearly indicates that the lanes will all be flowing at the same frequency.

CHAPTER FIVE
CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

A composite AI technique consisting of ANN and fuzzy logic have been applied in the development of an intelligent traffic management system. This has resulted to the development of a system that evenly distributes priorities to different lanes of a road making a traffic intersection, which is achieved by monitoring the traffic situation of the road using inexpensive and off-the-shelf technologies. However, the evenly distributed priorities resulted to optimal control of road traffic situation due to similar frequency at which traffic flows in different lanes within a period of time. Parameters such as the average waiting time of vehicles on traffic queues, number of vehicles on traffic queues, the frequency at which the traffic flows, and the duration of green light signal for each traffic phase were considered the key factors for determining the road traffic situation of a particular lane.

Nevertheless, techniques such as load balancing and RPC were applied to work in synchrony with the AI technique adopted. This facilitated ideal distribution of lane priorities and optimum interchange of information among the processes running on different microcontrollers coordinating the activities of the various lanes. Both the load balancer and RPC run on the lane traffic signal microcontrollers.

Although the system required certain cheap, programmable, and off-the-shelf hardware components such as the motion proximity sensors, microcontrollers, and LEDs; software program instructions (firmware) are required to drive the operational functions of these hardware. Object-oriented system modelling technique was applied for the analysis and design of the software programs as well as the entire system. By adopting object-oriented system modelling, the major entities that make up the system; their respective attributes, operations, and interrelationships were identified and clearly outlined by using Unified Modelling Language (UML) tools and patterns. This inherently expatiated data abstraction, encapsulation of implementation details, high cohesion, loose coupling, parallel programming of functions, and ease of maintainability; features that fostered the security and performance of the system. Rational Unified Process (RUP) was adopted by building, testing, and fixing the components of the system at regular intervals.

The system was simulated using Proteus 8.6, a Computer Aided Development suite. The simulation circuit was designed, and the code for priority checking of the lanes was configured in the Proteus 8.6 environment. With the use of the signal generators, the behaviour of the various sensors was emulated and the output was gotten from the microcontroller board using an oscilloscope. It is based on this that different lane signals were varied at different traffic phase in order to ascertain the even distribution of traffic flow.

5.2 Recommendations

In the future, the following recommendations can be applied to this piece of research in order to improve the general operations and performance of the system;

1. Different traffic signals situated at different locations should be made to interchange information among themselves in order to improve the intelligence of the system.
2. Data logging features and real-time clock module should be added to the system to enable it keep records of traffic situations at specific points in time. This will facilitate the prediction of traffic situations.

5.3 Contribution to Knowledge

This research developed an intelligent traffic management system utilizing DTAT that adopted a composite AI technique consisting of ANN and fuzzy logic for specifically monitoring and controlling road traffic situations at points where two or more road lanes meet. Also, it applied object-oriented system analysis in modelling and designing the system which gave rise to high cohesiveness, loose coupling, data abstraction, and encapsulation of system details. Furthermore, techniques such as load balancing and RPC were applied to work in synchrony with the AI technique adopted. This facilitated ideal distribution of lane priorities and the optimal control of road traffic.

References

Abdoos, M., & Bazzan, A. L. (2021). Hierarchical Traffic Signal Optimization using Reinforcement Learning and Traffic Prediction with Long-Short Term Memory. *Expert Syst. Appl*., *171*, 114580.

Abohashima, H., Gheith, M.S., & Eltawil, A.B. (2020). A proposed IoT based Smart traffic lights control system within a V2X framework. *2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, 338-343.

Abohashima, H., Gheith, M.S., & Eltawil, A.B. (2020). A proposed IoT based Smart traffic lights control system within a V2X framework. 2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES), 338-343.

Acharya, S., Dash, K. K., Chaini, R. (2020). Fuzzy Logic: An Advanced Approach to Traffic Control. In: Nayak, J., Balas, V., Favorskaya, M., Choudhury, B., Rao, S., Naik, B. (Eds.), *Applications of Robotics in Industry Using Advanced Mechanisms. Learning and Analytics in Intelligent Systems*, *Vol. 5*, (pp. 176-186). Springer, Cham. https://doi.org/10.1007/978-3-030-30271-9_17.

Adak, M. F., & Balta, H. (2021). Fuzzy Logic-based Adaptive Traffic Light Control of an Intersection: A Case Study. 5th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), (pp. 394-398). https://doi.org/10.1109/ISMSIT52890.2021.9604556.

Africa, A. D. (2019). Sensor-based Traffic Control Network with Neural Network Based Control System*. International Journal of Advanced Trends in Computer Science and Engineering. 8*, 983-989. 10.30534/ijatcse/2019/01842019.

Ajao, L. A., Ajao, F. J., Adegboye, M. A., & Ismail, A. A. (2018). An Embedded Fuzzy Logic Based Application for Density Traffic Control System. *International Journal of Artificial Intelligence Research*, *2*(1), 6-13.

Ali, M. E., Durdu, A., Celtek, S. A., & Gultekin, S. S. (2020). Fuzzy Logic and Webster's Optimal Cycle Based Decentralized Coordinated Adaptive Traffic Control Method. *Elektronika Ir Elektrotechnika*, *26*, 57-64.

Alomari, K., Mendoza, R., Sundermann, S., Goehring, D., & Rojas, R. (2020). Fuzzy Logic-based Adaptive Cruise Control for Autonomous Model Car. 121-130. 10.5220/0010175101210130.

Ariffin, W.N., Keat, C. S., Suriyan, T. P., Nore, N. A., Mohamad, I. S. M., Hasneeza, L. Z., Hashim, N. B., & Zain, A. S. (2021). Real-time Dynamic Traffic Light Control System with Emergency Vehicle Priority. *Journal of Physics: Conference Series, 1878*(1):012063.10.1088/1742-6596/1878/1/012063

Astarita, V., Festa, D. C., & Giofrè, V. P. (2018). Cooperative-Competitive Paradigm in Traffic Signal Synchronization Based on Floating Car Data. *IEEE Xplore*.

Ata, A., Khan, M.A., Abbas, S., Ahmad, G., & Fatima, A. (2019). Modelling Smart Road Traffic Congestion Control System Using Machine Learning Techniques. Neural Network World, (pp. 99-110). 10.14311/NNW.2019.29.008.

Awoyera, O.O., Sacko, O., Darboe, O., & Cynthia, O.C. (2019). Anfis-Based Intelligent Traffic Control System (ITCS) for Developing Cities. *Journal of Traffic and Logistics Engineering, 7(1), 18-22.*

Bagaa, M., Dutra, D. L. C, Taleb, T., & Samdanis, K. (2020). On SDN-Driven Network Optimization and QoS Aware Routing Using Multiple Paths. *IEEE Transactions on Wireless Communications*, *19*(7), 4700-4714.

Bharani, N. (2018). IoT Based Real Time Traffic Monitoring and Control System for Smart Cities. *IAETSD Journal for Advanced Research in Applied Sciences 5*(4), 2394-8442. http://iaetsdjaras.org/ 297

Casmir C. Onyeneke. (2018). Causes and Effects of Traffic Congestions in Nigeria. *Global Journal of Science Frontier Research*, *18*(F5), 7–16. Retrieved from https://journalofscience.org/index.php/GJSFR/article/view/2283

Celtek, S. A., Durdu, A., & Ali, M. E. (2021). Evaluating Action Durations for Adaptive Traffic Signal Control Based on Deep Q-Learning. *Int. J. Intell. Transp. Syst. Res.*, *19*, 557-571.

Chen, J., & Liu, H. (2021). Intelligent traffic control system based on open IoT and machine learning. *J. Intell. Fuzzy Syst.*, *40*, 7001-7012.

Chen, X., Lu, J., Zhao, J., Qu, Z., Yang, Y., & Xian, J. (2020). Traffic Flow Prediction at Varied Time Scales via Ensemble Empirical Mode Decomposition and Artificial Neural Network. Sustainability, 12, 3678.

Chen, X., Wu, S., Shi, C., Huang, Y., Yang, Y., Ke, R., & Zhao, J. (2020). Sensing Data Supported Traffic Flow Prediction via Denoising Schemes and ANN: A Comparison. *IEEE Sensors Journal*, *20*, 14317-14328.

Daeichian, A., & Haghani, A. (2019). Fuzzy Q-Learning-Based Multi-agent System for Intelligent Traffic Control by a Game Theory Approach. *Arabian Journal for Science and Engineering*, *43,* 3241-3247.

Daradkeh, Y. I., Kiri Saâdaoui ko, L. (2018). Development of QoS Methods in the Information Networks with Fractal Traffic. *Int'l Journal of Electronics and Telecommunications*, *64*(1), 27-32. https://doi.org/10.24425/118142.

Deulkar, V. (2019). Traffic Flow Control by using ANN Based Designing Toll Plaza Configuration. *International Journal of Scientific Research & Engineering Trends, 5*(4), 1412-1417.

Deveci, M., Pamučar, D., & Gokasar, I. (2021). Fuzzy Power Heronian function based CoCoSo method for the advantage prioritization of autonomous vehicles in real-time traffic management. Sustainable Cities and Society, 69, 102846.

Deveci, M., Pekaslan, D., & Canıtez, F. (2020). The assessment of smart city projects using zSlice type-2 fuzzy sets based Interval Agreement Method. Sustainable Cities and Society, 53, 101889.

Dey, M.R., Sharma, S., Shit, R.C., Meher, C.P., & Pati, H.K. (2019). IoV based Real-Time Smart Traffic Monitoring System for Smart Cities using Augmented Reality. 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), 1-6.

Dong, S., Yu, C., Yan, G., Zhu, J., & Hu, H. (2021). A Novel Ensemble Reinforcement Learning Gated Recursive Network for Traffic Speed Forecasting. 2021 Workshop on Algorithm and Big Data.

Dubey, A., Lakhani, M., Dave, S., & Patoliya, J. J. (2017). Internet of Things based adaptive traffic management system as a part of Intelligent Transportation System (ITS). *2017 International Conference on Soft Computing and its Engineering Applications (icSoftComp*), 1-6. Changa, ndia. 10.1109/ICSOFTCOMP.2017.8280081.

Ekedebe, N., Chen, Z., Xu, G., Lu, C., & Yu, W. (2014). On an efficient and effective Intelligent Transportation System (ITS) using field and simulation data. Sensing Technologies + Applications.

Garg, D., Chli, M., & Vogiatzis, G. (2018). Deep Reinforcement Learning for Autonomous Traffic Light Control. *2018 3rd IEEE International Conference on Intelligent Transportation Engineering* (*ICITE*), 214-218.

Garg, H., & Kaushal, E.G. (2017). Traffic Lights Control System for Indian Cities Using WSN and Fuzzy Control.

Gedik, A. (2020). Short-term Traffic Volume Prediction for the Merging Roads by Artificial Neural Network.

George, A. M., George, V. I., & George, M. A. (2018). IoT Based Smart Traffic Light Control System. *2018 International Conference on Control, Power, Communication and Computing Technologies (ICCPCCT*), 148-151.

Goudarzi, F., Asgari, H., & Al-Raweshidy, H. S. (2022). Traffic-Aware VANET Routing for City Environments - A Protocol Based on Ant Colony Optimization. *IEEE Systems Journal*, *13*(1), 571-581.

Goyal, M., Priya, A.V., Kumar, C.A., Verma, V., & Hota, M. (2018). An Ingenious Traffic Control System Using Fuzzy Logic Control with Emergency Override Feature. *2018 IEEE International Conference on System, Computation, Automation and Networking (ICSCA*), 1-6.

Guidoni, D. L., Maia, G., Souza, F, S. H., Villas, L. A., & Loureiro, A. A. F. (2020). Vehicular Traffic Management Based on Traffic Engineering for Vehicular Ad Hoc Networks. *IEEE Access*, *8*, 45167-45183. https://doi.org/10.1109/ACCESS.2020.2978700.

Gusrialdi, S., Qu, Z., & Simaan, M. A. (2018). Competitive Interaction Design of Cooperative Systems Against Attacks. *IEEE Transactions on Automatic Control*, 63(9), 3159-3166.

Hao, S., Yang, L., Ding, L., & Guo, Y. (2019). Distributed Cooperative Backpressure-Based Traffic Light Control Method. *Journal of Advanced Transportation*, 1-14.

Ikidid, A., & Abdelaziz, E.F. (2019). Multi-Agent and Fuzzy Inference Based Framework for Urban Traffic Simulation. *2019 International Conference on Systems of Collaboration Big Data, Internet of Things & Security (SysCoBIoTS)*, 1-6.

Ikidid, A., Fazziki, A.E., & Sadgal, M. (2021). A Fuzzy Logic Supported Multi-Agent System For Urban Traffic And Priority Link Control. *J. Univers. Comput. Sci.*, *27*, 1026-1045.

Jahnavi, S., Prasanth, G., Priyanka, D., Sneheth, A., & Navya, M. (2021). Intelligent Traffic Light Management System. In: Kiran, M. C., Kiranmayee, B. V., Favorskaya, M. N., Chandra, S. S., & Raju K. S. (eds). Proceedings of International Conference on Advances in Computer Engineering and

Communication Systems. *Learning and Analytics in Intelligent Systems*, 20. https://doi.org/10.1007/978-981-15-9293-5_45.

Jaiswal, K., & Anand, V. (2020). EOMR: An Energy-Efficient Optimal Multipath Routing Protocol to Improve QoS in Wireless Sensor Network for IoT Applications. *Wireless Personal Communications*, 111, 2493–2515.

Jaiswal, K., & Anand, V. (2020). EOMR: An Energy-Efficient Optimal Multi-Path Routing Protocol to Improve QoS in Wireless Sensor Network for IoT Applications. *Wireless Personal Communications, 111*, 2493-2515.

Karmakar, G. C., Chowdhury, A., Kamruzzaman, J., & Gondal, I. (2021). A Smart Priority-Based Traffic Control System for Emergency Vehicles. *IEEE Sensors Journal, 21*, 15849-15858.

Kawade, D., Deshmukh, S., Gamare, S., Sankhe, A. (2018). Smart Traffic Control Using OpenCV. *IOSR Journal of. Engineering. (IOSRJEN*).

Khattak, H., Farman, H., Jan, B., & Din, I.U. (2019). Toward Integrating Vehicular Clouds with IoT for Smart City Services. *IEEE Network*, *33*, 65-71.

Kukic, K., & Jovanović, A. (2019). Fuzzy logic approach on traffic control of a Diverging Diamond Interchange in real time. *ITM Web of Conferences*.

Kumar, N., Rahman, S. S., & Dhakad, N. (2021). Fuzzy Inference Enabled Deep Reinforcement Learning-Based Traffic Light Control for Intelligent Transportation System. *IEEE Transactions on Intelligent Transportation Systems*, *22*(8), 4919-4928.

Latif, A., & Megantoro, P. (2020). Traffic Light Regulatory System Based on Fuzzy Algorithm Using Microcontroller. Journal of Physics: Conference Series, *1464*,

Latif, S., Afzaal, H., & Zafar, N. A. (2018). Intelligent Traffic Monitoring and Guidance System for Smart City. *International Conference on Computing, Mathematics and Engineering Technologies*-iCoMET 2018. https://doi.org/978-1-5386-1370-2/18/$31.00 ©2018 IEEE.

Latif, S., Afzaal, H., & Zafar, N.A. (2018). Intelligent traffic monitoring and guidance system for smart city. 2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), (pp. 1-6).

Lau, C.Y., Yuen, M., Yueng, K., Fan, C., Ko, O., Ngan, L., Tam, W., & Yeung, W. (2022). PC-based Intelligent Traffic Monitoring System with Real-time Analysis for Smart Cities. 2022 14th International Conference on COMmunication Systems & NETworkS (COMSNETS), (pp. 324-328).

Lee, S., Kim, Y., Kahng, H., Lee, S., Chung, S., Cheong, T., Shin, K., Park, J., & Kim, S.B. (2020). Intelligent traffic control for autonomous vehicle systems based on machine learning. *Expert Syst. Appl., 144*, 113074.

Liang, X., Du, X., Wang, G., & Han, Z. (2018). A Deep Reinforcement Learning Network for Traffic Light Cycle Control. IEEE Transactions on Vehicular Technology, 68, 1243-1253.

Lima, G. (2019). Embedded System for Real-Time Traffic Monitoring on Smart Cities.

Lin, Y., Dai, X., Li, L., & Wang, F. (2019). Pattern Sensitive Prediction of Traffic Flow Based on Generative Adversarial Framework. *IEEE Transactions on Intelligent Transportation Systems, 20*(6), 2395-2400.

Louati, A., Louati, H., & Li, Z. (2021). Deep learning and case-based reasoning for predictive and adaptive traffic emergency management. *J. Supercomput., 77*, 4389-4418.

Lu, Y., Liu, J., Sivaparthipan, C.B., & Marimuthu, K. (2021). Correction to: Context-Aware Evaluation of HCI Systems for Transportation in Urban Areas. *Arabian Journal for Science and Engineering. https://doi.org/10.1007/s13369-021-05986-w*.

Mohammadi, R., Javidan, R., Keshtgari, M., & Akbari, R. (2018). A novel multicast traffic engineering technique in SDN using TLBO algorithm. *Telecommunication Systems, 68*, 583-592.

Mubarak, Y.D., Kirichenko, L., & Radivilova, T. (2018). Development of QoS methods in the information networks with fractal traffic. *ArXiv, abs/1904.05202.*

Muthuramalingam, S., Bharathi, A., Kumar, S.R., Gayathri, N.B., Sathiyaraj, R., & Balamurugan, B. (2018). IoT Based Intelligent Transportation System (IoT-ITS) for Global Perspective: A Case Study. Intelligent Systems Reference Library.

Nagaraju, D.R., Reddy, M.K., Harika, M., Reddy, N.S., & Deepthi, P. (2019). An Improved Traffic Congestion Control and Vehicle Monitoring System.

Ng, S.C., & Kwok, C. (2020). An Intelligent Traffic Light System Using Object Detection and Evolutionary Algorithm for Alleviating Traffic Congestion in Hong Kong. *Int. J. Comput. Intell. Syst., 13*, 802-809.

Nguyen, H., Kieu, L.M., Wen, T., & Cai, C. (2018). Deep learning methods in transportation domain: a review. *IET Intelligent Transport Systems*.

Pahal, N., Goel, D., & Chaudhury, S. (2018). Environment Monitoring System for Smart Cities Using Ontology. *Innovations in Smart Cities Applications Edition 2.*

Putra, A.S., & Warnars, H.L. (2018). Intelligent Traffic Monitoring System (ITMS) for Smart City Based on IoT Monitoring. 2018 *Indonesian Association for Pattern Recognition International Conference (INAPR)*, 161-165.

Rehman, A., Mazhar Rathore, M., Paul, A., Saeed, F., & Ahmad, R.W. (2018). Vehicular traffic optimisation and even distribution using ant colony in smart city environment. *IET Intelligent Transport Systems*.

Rida, N., Ouadoud, M., Hasbi, A., & Chebli, S. (2018). Adaptive Traffic Light Control System Using Wireless Sensors Networks. *2018 IEEE 5th International Congress on Information Science and Technology (CiSt)*, 552-556.

Rodríguez, F.L., Dias, U.S., Campelo, D.R., Albuquerque, R.D., Lim, S., & Garcia-Villalba, L.J. (2019). QoS Management and Flexible Traffic Detection Architecture for 5G Mobile Networks. Sensors (Basel, Switzerland), 19.

Romero-Ania, A., Rivero Gutiérrez, L., & De Vicente Oliva, M.A. (2021). Multiple Criteria Decision Analysis of Sustainable Urban Public Transport Systems. *Mathematics.*

Saâdaoui, F., Saadaoui, H., & Rabbouch, H. (2019). Hybrid feedforward ANN with NLS-based regression curve fitting for US air traffic forecasting. *Neural Computing and Applications, 32*, 10073-10085.

Saha, N., Bera, S., Misra, S. (2018). Sway: Traffic-aware QoS Routing in Software-Defined IoT. *IEEE Transactions on Emerging Topics in Computing*, 9 https://doi.org/10.1109/TETC.2018.2847296.

Sahraei, M.A. (2018). Modelling of traffic control delays at priority junctions using artificial neural network.

Sahraei, M.A., & Puan, O.C. (2018). Traffic Delay Estimation Using Artificial Neural Network (ANN) at Unsignalized Intersections. *Proceedings of the 3rd International Conference on Civil, Structural and Transportation Engineering (ICCSTE'18).*

Sharma, B., Kumar, S., Tiwari, P., Yadav, P., & Nezhurina, M.I. (2018). ANN based short-term traffic flow forecasting in undivided two lane highway. *Journal of Big Data, 5,* 1-16.

Sliwa, B., Liebig, T., Vranken, T., Schreckenberg, M., & Wietfeld, C. (2019). System-of-Systems Modeling, Analysis and Optimization of Hybrid Vehicular Traffic. *2019 IEEE International Systems Conference (SysCon),* 1-8.

Song, Z., Guo, Y., Wu, Y., & Ma, J. (2019). Short-term traffic speed prediction under different data collection time intervals using a SARIMA-SDGM hybrid prediction model. *PLoS ONE*, 14.

Sopeña, J.M., Pakrashi, V., & Ghosh, B. (2021). Interval prediction for short-term traffic forecasting using hybrid mode decomposition models. *2021 IEEE International Intelligent Transportation Systems Conference (ITSC),* 3246-3251.

Sridhar, S., Fathima, S.M., Vani, S.B., Nanditha, G.T., & Anusha, K. (2019). Intelligent Traffic Management System Based on Internet of Vehicles (IOV).

Sumalatha, V., Dingari, M., & Jayalakshmi, C. (2019). Short Term Road Traffic Flow Forecasting using Artificial Neural Network. *International Journal of Recent Technology and Engineering*, 8(3), 7998-8000.

Tan, D., Younis, M.F., Lalouani, W., & Lee, S. (2021). PALM: Platoons Based Adaptive Traffic Light Control System for Mixed Vehicular Traffic. *2021 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/IOP/SCI),* 178-185.

Tchappi, I.H., Galland, S., Kamla, V.C., Kamgang, J.C., Mualla, Y., Najjar, A., & Hilaire, V. (2020). A critical review of the use of holonic paradigm in traffic and transportation systems. *Eng. Appl. Artif. Intell., 90*, 103503.

UKEssays. (2018). “Causes and Effects of Traffic Congestion”. Retrieved from https://www.ukessays.com/essays/tourism/causes-and-effects-of-traffic-congestion-tourism-essay.php?vref=1

Varyani, N., Zhang, Z., & Dai, D.H. (2020). QROUTE: An Efficient Quality of Service (QoS) Routing Scheme for Software-Defined Overlay Networks. *IEEE Access, 8*, 104109-104126.

Viloria, A., Lezama, O. B., Varela, N., & Martinez, J. L. (2020). RETRACTED CHAPTER: Optimization of Driving Efficiency for Pre-determined Routes:

Proactive Vehicle Traffic Control. *Computing Science, Communication and Security, 1235*, 82 - 94.

Vogel, A., Oremović, I., Šimić, R., & Ivanjko, E. (2018). Improving Traffic Light Control by Means of Fuzzy Logic. *2018 International Symposium ELMAR*, 51-56.

Wang, X., Ning, Z., Hu, X., Ngai, E.C., Wang, L., Hu, B., & Kwok, Y. (2018). A City-Wide Real-Time Traffic Management System: Enabling Crowdsensing in Social Internet of Vehicles. *IEEE Communications Magazine, 56*, 19-25.

Wang, Y., Yang, X., Liang, H., & Liu, Y. (2018). A Review of the Self-Adaptive Traffic Signal Control System Based on Future Traffic Environment. *Journal of Advanced Transportation.*

Won, M. (2020). Intelligent Traffic Monitoring Systems for Vehicle Classification: A Survey. *IEEE Access*, *8*, 73340-72258. https://doi.org/10.1109/ACCESS.2020.2987634

Xu, M., An, K., Vu, H. L., Ye, Z., Feng, J., & Chen, E. (2019). Optimizing multi-agent based urban traffic signal control system. *Journal of Intelligent Transportation Systems, 23*, 357 - 369.

Yang, S., Xu, C., Zhong, L., Shen, J., & Muntean, G. (2020). A QoE-Driven Multicast Strategy with Segment Routing-A Novel Multimedia Traffic Engineering Paradigm. *IEEE Transactions on Broadcasting, 66*, 34-46.

Zadobrischi, E., & Dimian, M. (2021). Vehicular Communications Utility in Road Safety Applications: A Step toward Self-Aware Intelligent Traffic Systems. *Symmetry, 13*, 438.

Zhang, Z., & Zhang, W. (2020). Traffic Pattern Analysis and Traffic State Prediction of Urban Traffic Road Network Based on Correlated Routes. *Proceedings of the 39th Chinese Control Conference.* 5654-5659.

Zheng, G., Chai, W.K., & Katos, V. (2019). An Ensemble Model for Short-Term Traffic Prediction in Smart City Transportation System. *2019 IEEE Global Communications Conference (GLOBECOM)*, 1-6.

APPENDIX 1: Program Code for Load Balancing

```
require('babel-register');
require('babel-polyfill');
module.exports = {
  development: {
"nbformat": 4,
 "nbformat_minor": 0,
 "metadata": {
   "anaconda-cloud": {},
   "kernelspec": {
     "name": "python3",
     "display_name": "Python 3"
   },
   "colab": {
     "name": "TLCS.ipynb",
     "provenance": [],
     "collapsed_sections": [],
     "toc_visible": true
   },
   "accelerator": "TPU"
 },
 "cells": [
   {
     "cell_type": "markdown",
     "metadata": {
```

"id": "nhde1biSjajW"
},
"source": [
"# **Setup for TLCS**"
]
},
{
"cell_type": "code",
"metadata": {
"id": "VZZSh853VCpT",
"outputId": "c2d6c77b-3e5f-4c18-eb49-00a8ab491a97",
"colab": {
"base_uri": "https://localhost:8080/",
"height": 51
}
},
"source": [
"!pwd\n",
"!ls\n",
"!rm -r *\n",
"!ls"
],
"execution_count": 1,
"outputs": [
{

```
      "output_type": "stream",
      "text": [
       "/content\n",
       "sample_data\n"
      ],
      "name": "stdout"
     }
    ]
   },
   {
    "cell_type": "code",
    "metadata": {
     "id": "GA53_oDvE8eN",
     "outputId": "6638e416-6098-42ab-d80b-9d8112bf5ac0",
     "colab": {
      "base_uri": "https://localhost:8080/",
      "height": 102
     }
    },
    "source": [
     "!git clone \"https://github.com/AndreaVidali/Deep-QLearning-Agent-for-Traffic-Signal-Control\""
    ],
    "execution_count": 2,
    "outputs": [
     {
```

"output_type": "stream",
"text": [
"Cloning into 'Deep-QLearning-Agent-for-Traffic-Signal-Control'...\n",
"remote: Enumerating objects: 168, done.\u001b[K\n",
"remote: Total 168 (delta 0), reused 0 (delta 0), pack-reused 168\u001b[K\n",
"Receiving objects: 100% (168/168), 156.58 KiB | 2.06 MiB/s, done.\n",
"Resolving deltas: 100% (81/81), done.\n"
],
"name": "stdout"
}
]
},
{
"cell_type": "code",
"metadata": {
"id": "ZzlPMXYpFXi9"
},
"source": [
"!mv /content/Deep-QLearning-Agent-for-Traffic-Signal-Control/TLCS/* /content/\n",
"!rm -r Deep-QLearning-Agent-for-Traffic-Signal-Control"
],
"execution_count": 3,
"outputs": []
},
{

"cell_type": "code",
"metadata": {
  "id": "MVyz7OCYpA85",
  "outputId": "1b681078-7a39-4f69-f584-9116da2dacc8",
  "colab": {
    "base_uri": "https://localhost:8080/",
    "height": 238
  }
},
"source": [
  "!ls -l"
],
"execution_count": 4,
"outputs": [
  {
    "output_type": "stream",
    "text": [
      "total 72\n",
      "-rw-r--r-- 1 root root 5157 Sep 30 22:26 generator.py\n",
      "drwxr-xr-x 2 root root 4096 Sep 30 22:26 intersection\n",
      "-rw-r--r-- 1 root root   978 Sep 30 22:26 memory.py\n",
      "-rw-r--r-- 1 root root 3312 Sep 30 22:26 model.py\n",
      "-rw-r--r-- 1 root root 1752 Sep 30 22:26 testing_main.py\n",
      "-rw-r--r-- 1 root root   258 Sep 30 22:26 testing_settings.ini\n",
      "-rw-r--r-- 1 root root 8692 Sep 30 22:26 testing_simulation.py\n",

```
      "-rw-r--r-- 1 root root  2794 Sep 30 22:26 training_main.py\n",
      "-rw-r--r-- 1 root root   413 Sep 30 22:26 training_settings.ini\n",
      "-rw-r--r-- 1 root root 11990 Sep 30 22:26 training_simulation.py\n",
      "-rw-r--r-- 1 root root  4658 Sep 30 22:26 utils.py\n",
      "-rw-r--r-- 1 root root  1006 Sep 30 22:26 visualization.py\n"
     ],
     "name": "stdout"
    }
   ]
  },
  {
   "cell_type": "code",
   "metadata": {
    "id": "8zjb7bq_jrSp",
    "outputId": "4143283a-bf3b-4ab2-acb7-6c61aea55d0c",
    "colab": {
     "base_uri": "https://localhost:8080/",
     "height": 34
    }
   },
   "source": [
    "!python --version"
   ],
   "execution_count": null,
   "outputs": [
```

{
"output_type": "stream",
"text": [
"Python 3.6.9\n"
],
"name": "stdout"
}
]
},
{
"cell_type": "markdown",
"metadata": {
"id": "CZ0Ptg7fkXrY"
},
"source": [
"Installing **SUMO TraCI**"
]
},
{
"cell_type": "code",
"metadata": {
"id": "3Oqs5WNyVpco",
"outputId": "d5f5a6cf-da87-4d12-a2c4-0c4d052f0d06",
"colab": {
"base_uri": "https://localhost:8080/",

"height": 1000
}
},
"source": [
"!add-apt-repository ppa:sumo/stable -y\n",
"!apt-get update -y\n",
"!apt-get install sumo sumo-tools sumo-doc"
],
"execution_count": null,
"outputs": [
{
"output_type": "stream",
"text": [
"\r0% [Working]\r \rGet:1 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]\n",
"\r0% [Connecting to archive.ubuntu.com (91.189.88.142)] [1 InRelease 14.2 kB/88.7\r \rGet:2 https://cloud.r-project.org/bin/linux/ubuntu bionic-cran35/ InRelease [3,626 B]\n",
"\r0% [Connecting to archive.ubuntu.com (91.189.88.142)] [1 InRelease 14.2 kB/88.7\r0% [Connecting to archive.ubuntu.com (91.189.88.142)] [1 InRelease 43.1 kB/88.7\r0% [2 InRelease gpgv 3,626 B] [Connecting to archive.ubuntu.com (91.189.88.142)\r \rIgn:3 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 InRelease\n",
"\r0% [2 InRelease gpgv 3,626 B] [Waiting for headers] [1 InRelease 43.1 kB/88.7 k\r0% [2 InRelease gpgv 3,626 B] [Waiting for headers] [Waiting for headers] [Wait\r \rIgn:4 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 InRelease\n",
"Get:5 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 Release [697 B]\n",
"Hit:6 http://archive.ubuntu.com/ubuntu bionic InRelease\n",

"Hit:7 http://ppa.launchpad.net/graphics-drivers/ppa/ubuntu bionic InRelease\n",

"Get:8 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 Release [564 B]\n",

"Get:9 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 Release.gpg [801 B]\n",

"Get:10 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 Release.gpg [833 B]\n",

"Get:11 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]\n",

"Get:12 http://ppa.launchpad.net/marutter/c2d4u3.5/ubuntu bionic InRelease [15.4 kB]\n",

"Get:13 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [961 kB]\n",

"Get:14 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]\n",

"Get:15 http://ppa.launchpad.net/sumo/stable/ubuntu bionic InRelease [15.4 kB]\n",

"Ign:16 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 Packages\n",

"Get:16 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 Packages [151 kB]\n",

"Get:17 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 Packages [38.7 kB]\n",

"Get:18 http://ppa.launchpad.net/marutter/c2d4u3.5/ubuntu bionic/main Sources [1,840 kB]\n",

"Get:19 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [1,397 kB]\n",

"Get:20 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [1,254 kB]\n",

"Get:21 http://ppa.launchpad.net/marutter/c2d4u3.5/ubuntu bionic/main amd64 Packages [888 kB]\n",

"Get:22 http://ppa.launchpad.net/sumo/stable/ubuntu bionic/main amd64 Packages [901 B]\n",

"Fetched 6,819 kB in 4s (1,813 kB/s)\n",

"Reading package lists... Done\n",

"Hit:1 https://cloud.r-project.org/bin/linux/ubuntu bionic-cran35/ InRelease\n",

"Hit:2 http://security.ubuntu.com/ubuntu bionic-security InRelease\n", "Ign:3 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 InRelease\n",

"Ign:4 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 InRelease\n",

"Hit:5 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 Release\n",

"Hit:6 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 Release\n",

"Hit:7 http://archive.ubuntu.com/ubuntu bionic InRelease\n",

"Hit:8 http://ppa.launchpad.net/graphics-drivers/ppa/ubuntu bionic InRelease\n",

"Hit:9 http://archive.ubuntu.com/ubuntu bionic-updates InRelease\n",

"Hit:10 http://archive.ubuntu.com/ubuntu bionic-backports InRelease\n",

"Hit:11 http://ppa.launchpad.net/marutter/c2d4u3.5/ubuntu bionic InRelease\n",

"Hit:12 http://ppa.launchpad.net/sumo/stable/ubuntu bionic InRelease\n",

"Reading package lists... Done\n",

"Reading package lists... Done\n",

"Building dependency tree   \n",

"Reading state information... Done\n",

"The following package was automatically installed and is no longer required:\n",

" libnvidia-common-440\n",

"Use 'apt autoremove' to remove it.\n",

"The following additional packages will be installed:\n",

" binfmt-support fastjar jarwrapper javascript-common libfox-1.6-0\n",

" libjs-jquery libjs-openlayers libjs-underscore\n",

"Suggested packages:\n",

" apache2 | lighttpd | httpd\n",

"The following NEW packages will be installed:\n",

" binfmt-support fastjar jarwrapper javascript-common libfox-1.6-0\n",

" libjs-jquery libjs-openlayers libjs-underscore sumo sumo-doc sumo-tools\n",

"0 upgraded, 11 newly installed, 0 to remove and 53 not upgraded.\n",

"Need to get 25.6 MB of archives.\n",

"After this operation, 154 MB of additional disk space will be used.\n",

"Get:1 http://archive.ubuntu.com/ubuntu bionic/main amd64 binfmt-support amd64 2.1.8-2 [51.6 kB]\n",

"Get:2 http://ppa.launchpad.net/sumo/stable/ubuntu bionic/main amd64 sumo amd64 1.6.0+dfsg1-1 [12.0 MB]\n",

"Get:3 http://archive.ubuntu.com/ubuntu bionic/universe amd64 fastjar amd64 2:0.98-6build1 [66.7 kB]\n",

"Get:4 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 jarwrapper all 0.72.1~18.04.1 [20.3 kB]\n",

"Get:5 http://archive.ubuntu.com/ubuntu bionic/main amd64 javascript-common all 11 [6,066 B]\n",

"Get:6 http://archive.ubuntu.com/ubuntu bionic/universe amd64 libfox-1.6-0 amd64 1.6.56-1 [875 kB]\n",

"Get:7 http://archive.ubuntu.com/ubuntu bionic/main amd64 libjs-jquery all 3.2.1-1 [152 kB]\n",

"Get:8 http://archive.ubuntu.com/ubuntu bionic/main amd64 libjs-underscore all 1.8.3~dfsg-1 [59.9 kB]\n",

"Get:9 http://archive.ubuntu.com/ubuntu bionic/universe amd64 libjs-openlayers all 2.13.1+ds2-4 [704 kB]\n",

"Get:10 http://ppa.launchpad.net/sumo/stable/ubuntu bionic/main amd64 sumo-doc all 1.6.0+dfsg1-1 [787 kB]\n",

"Get:11 http://ppa.launchpad.net/sumo/stable/ubuntu bionic/main amd64 sumo-tools all 1.6.0+dfsg1-1 [10.9 MB]\n",

"Fetched 25.6 MB in 4s (6,822 kB/s)\n",

"Selecting previously unselected package binfmt-support.\n",

"(Reading database ... 144328 files and directories currently installed.)\n", "Preparing to unpack .../00-binfmt-support_2.1.8-2_amd64.deb ...\n",

"Unpacking binfmt-support (2.1.8-2) ...\n",

"Selecting previously unselected package fastjar.\n",

"Preparing to unpack .../01-fastjar_2%3a0.98-6build1_amd64.deb ...\n",

"Unpacking fastjar (2:0.98-6build1) ...\n",

"Selecting previously unselected package jarwrapper.\n",

"Preparing to unpack .../02-jarwrapper_0.72.1~18.04.1_all.deb ...\n",

"Unpacking jarwrapper (0.72.1~18.04.1) ...\n",

"Selecting previously unselected package javascript-common.\n",

"Preparing to unpack .../03-javascript-common_11_all.deb ...\n",

"Unpacking javascript-common (11) ...\n",

"Selecting previously unselected package libfox-1.6-0:amd64.\n",

"Preparing to unpack .../04-libfox-1.6-0_1.6.56-1_amd64.deb ...\n",

"Unpacking libfox-1.6-0:amd64 (1.6.56-1) ...\n",

"Selecting previously unselected package libjs-jquery.\n",

"Preparing to unpack .../05-libjs-jquery_3.2.1-1_all.deb ...\n",

"Unpacking libjs-jquery (3.2.1-1) ...\n",

"Selecting previously unselected package libjs-underscore.\n",

"Preparing to unpack .../06-libjs-underscore_1.8.3~dfsg-1_all.deb ...\n",

"Unpacking libjs-underscore (1.8.3~dfsg-1) ...\n",

"Selecting previously unselected package sumo.\n",

"Preparing to unpack .../07-sumo_1.6.0+dfsg1-1_amd64.deb ...\n",

"Unpacking sumo (1.6.0+dfsg1-1) ...\n",

"Selecting previously unselected package sumo-doc.\n",

"Preparing to unpack .../08-sumo-doc_1.6.0+dfsg1-1_all.deb ...\n", "Unpacking sumo-doc (1.6.0+dfsg1-1) ...\n",

"Selecting previously unselected package libjs-openlayers.\n",

"Preparing to unpack .../09-libjs-openlayers_2.13.1+ds2-4_all.deb ...\n",

"Unpacking libjs-openlayers (2.13.1+ds2-4) ...\n",

"Selecting previously unselected package sumo-tools.\n",

"Preparing to unpack .../10-sumo-tools_1.6.0+dfsg1-1_all.deb ...\n",

"Unpacking sumo-tools (1.6.0+dfsg1-1) ...\n",

"Setting up libjs-jquery (3.2.1-1) ...\n",

"Setting up binfmt-support (2.1.8-2) ...\n",

"Created symlink /etc/systemd/system/multi-user.target.wants/binfmt-support.service ? /lib/systemd/system/binfmt-support.service.\n",

"invoke-rc.d: could not determine current runlevel\n",

"invoke-rc.d: policy-rc.d denied execution of start.\n",

"Setting up libfox-1.6-0:amd64 (1.6.56-1) ...\n",

"Setting up libjs-underscore (1.8.3~dfsg-1) ...\n",

"Setting up fastjar (2:0.98-6build1) ...\n",

"Setting up libjs-openlayers (2.13.1+ds2-4) ...\n",

"Setting up sumo (1.6.0+dfsg1-1) ...\n",

"Setting up javascript-common (11) ...\n",

"Setting up jarwrapper (0.72.1~18.04.1) ...\n",

"Setting up sumo-doc (1.6.0+dfsg1-1) ...\n",

"Setting up sumo-tools (1.6.0+dfsg1-1) ...\n",

"Processing triggers for systemd (237-3ubuntu10.41) ...\n",

"Processing triggers for man-db (2.8.3-2ubuntu0.1) ...\n",

"Processing triggers for mime-support (3.60ubuntu1) ...\n",

"Processing triggers for libc-bin (2.27-3ubuntu1) ...\n",

"/sbin/ldconfig.real: /usr/local/lib/python3.6/dist-packages/ideep4py/lib/libmkldnn.so.0 is not a symbolic link\n",
"\n"
],
"name": "stdout"
}
]
},
{
"cell_type": "code",
"metadata": {
"id": "AkoK63N3mR4E",
"outputId": "f0efee91-beb1-48cc-b1db-e502d07d27c0",
"colab": {
"base_uri": "https://localhost:8080/",
"height": 173
}
},
"source": [
"!pip install traci"
],
"execution_count": null,
"outputs": [
{
"output_type": "stream",
"text": [

"Collecting traci\n",

"\u001b[?25l Downloading https://files.pythonhosted.org/packages/e6/93/3c68e9f61c8415bf727ae3ac48627c54e4141d284ba0b37c5da810939ca6/traci-1.6.0-py3-none-any.whl (206kB)\n",

"\u001b[K | | 215kB 3.5MB/s \n",

"\u001b[?25hCollecting sumolib>=1.6.0\n",

"\u001b[?25l Downloading https://files.pythonhosted.org/packages/67/f2/c026667c006420cf98948ec3a81f4a68da2f386e14c323ad917462a74267/sumolib-1.6.0-py3-none-any.whl (130kB)\n",

"\u001b[K | | 133kB 10.6MB/s \n",

"\u001b[?25hInstalling collected packages: sumolib, traci\n",

"Successfully installed sumolib-1.6.0 traci-1.6.0\n"

],

"name": "stdout"

}

]

},

{

"cell_type": "code",

"metadata": {

"id": "s-6v3Jk1oxwE"

},

"source": [

"import os\n",

"os.environ['SUMO_HOME'] = \"/usr/share/sumo/\""

],

"execution_count": null,

"outputs": []
},
{
"cell_type": "markdown",
"metadata": {
"id": "ebzkYv4_klSV"
},
"source": [
"**Starting Training Process**"
]
},
{
"cell_type": "code",
"metadata": {
"id": "b2ACqSSajWoT",
"outputId": "031b34d7-bca5-4eba-8700-3ff292fd9c92",
"colab": {
"base_uri": "https://localhost:8080/",
"height": 1000
}
},
"source": [
"!python training_main.py"
],
"execution_count": null,

"outputs": [
{
"output_type": "stream",
"text": [
"2020-06-23 15:09:58.148129: E tensorflow/stream_executor/cuda/cuda_driver.cc:313] failed call to cuInit: CUDA_ERROR_NO_DEVICE: no CUDA-capable device is detected\n",
"\n",
"----- Episode 1 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -24710.0 - Epsilon: 1.0\n",
"Training...\n",
"Simulation time: 8.4 s - Training time: 0.0 s - Total: 8.4 s\n",
"\n",
"----- Episode 2 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -26416.0 - Epsilon: 0.99\n",
"Training...\n",
"Simulation time: 8.2 s - Training time: 101.4 s - Total: 109.6 s\n",
"\n",
"----- Episode 3 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",

```
"Simulating...\n",
"Total reward: -35112.0 - Epsilon: 0.98\n",
"Training...\n",
"Simulation time: 8.6 s - Training time: 102.2 s - Total: 110.8 s\n",
"\n",
"----- Episode 4 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -31930.0 - Epsilon: 0.97\n",
"Training...\n",
"Simulation time: 8.7 s - Training time: 100.4 s - Total: 109.1 s\n",
"\n",
"----- Episode 5 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -22242.0 - Epsilon: 0.96\n",
"Training...\n",
"Simulation time: 8.4 s - Training time: 102.8 s - Total: 111.2 s\n",
"\n",
"----- Episode 6 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
```

"Total reward: -34009.0 - Epsilon: 0.95\n",

"Training...\n",

"Simulation time: 9.4 s - Training time: 105.9 s - Total: 115.3 s\n",

"\n",

"----- Episode 7 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -17112.0 - Epsilon: 0.94\n",

"Training...\n",

"Simulation time: 8.4 s - Training time: 102.4 s - Total: 110.8 s\n",

"\n",

"----- Episode 8 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -20555.0 - Epsilon: 0.93\n",

"Training...\n",

"Simulation time: 9.1 s - Training time: 101.4 s - Total: 110.5 s\n",

"\n",

"----- Episode 9 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -19614.0 - Epsilon: 0.92\n",

```
"Training...\n",
"Simulation time: 8.8 s - Training time: 100.6 s - Total: 109.4 s\n",
"\n",
"----- Episode 10 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -23736.0 - Epsilon: 0.91\n",
"Training...\n",
"Simulation time: 8.8 s - Training time: 99.3 s - Total: 108.1 s\n",
"\n",
"----- Episode 11 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -24066.0 - Epsilon: 0.9\n",
"Training...\n",
"Simulation time: 9.2 s - Training time: 99.2 s - Total: 108.4 s\n",
"\n",
"----- Episode 12 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -25121.0 - Epsilon: 0.89\n",
"Training...\n",
```

"Simulation time: 9.0 s - Training time: 100.2 s - Total: 109.2 s\n",

"\n",

"----- Episode 13 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -16412.0 - Epsilon: 0.88\n",

"Training...\n",

"Simulation time: 9.7 s - Training time: 99.9 s - Total: 109.6 s\n",

"\n",

"----- Episode 14 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -19786.0 - Epsilon: 0.87\n",

"Training...\n",

"Simulation time: 9.0 s - Training time: 100.6 s - Total: 109.6 s\n",

"\n",

"----- Episode 15 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -16815.0 - Epsilon: 0.86\n",

"Training...\n",

"Simulation time: 9.1 s - Training time: 99.7 s - Total: 108.8 s\n",

"\n",

"----- Episode 16 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -30985.0 - Epsilon: 0.85\n",

"Training...\n",

"Simulation time: 9.7 s - Training time: 100.7 s - Total: 110.4 s\n",

"\n",

"----- Episode 17 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -18319.0 - Epsilon: 0.84\n",

"Training...\n",

"Simulation time: 9.3 s - Training time: 100.5 s - Total: 109.8 s\n",

"\n",

"----- Episode 18 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -19557.0 - Epsilon: 0.83\n",

"Training...\n",

"Simulation time: 9.4 s - Training time: 101.0 s - Total: 110.4 s\n",

"\n",

```
"----- Episode 19 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -18599.0 - Epsilon: 0.82\n",
"Training...\n",
"Simulation time: 9.4 s - Training time: 100.4 s - Total: 109.8 s\n",
"\n",
"----- Episode 20 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -15339.0 - Epsilon: 0.81\n",
"Training...\n",
"Simulation time: 9.8 s - Training time: 100.2 s - Total: 110.0 s\n",
"\n",
"----- Episode 21 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -20444.0 - Epsilon: 0.8\n",
"Training...\n",
"Simulation time: 10.2 s - Training time: 99.8 s - Total: 110.0 s\n",
"\n",
"----- Episode 22 of 100\n",
```

```
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -16207.0 - Epsilon: 0.79\n",
"Training...\n",
"Simulation time: 10.4 s - Training time: 101.3 s - Total: 111.7 s\n",
"\n",
"----- Episode 23 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Warning: Choosing new speed factor 1.25 for vehicle 'N_E_255' to match departure speed.\n",
"Total reward: -15486.0 - Epsilon: 0.78\n",
"Training...\n",
"Simulation time: 10.0 s - Training time: 100.0 s - Total: 110.0 s\n",
"\n",
"----- Episode 24 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -14845.0 - Epsilon: 0.77\n",
"Training...\n",
"Simulation time: 10.6 s - Training time: 101.0 s - Total: 111.6 s\n",
"\n",
"----- Episode 25 of 100\n",
```

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -15499.0 - Epsilon: 0.76\n",

"Training...\n",

"Simulation time: 10.8 s - Training time: 101.7 s - Total: 112.5 s\n",

"\n",

"----- Episode 26 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -14869.0 - Epsilon: 0.75\n",

"Training...\n",

"Simulation time: 10.8 s - Training time: 103.6 s - Total: 114.4 s\n",

"\n",

"----- Episode 27 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -13010.0 - Epsilon: 0.74\n",

"Training...\n",

"Simulation time: 10.9 s - Training time: 102.6 s - Total: 113.5 s\n",

"\n",

"----- Episode 28 of 100\n",

" Retrying in 1 seconds\n",

```
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -16417.0 - Epsilon: 0.73\n",
"Training...\n",
"Simulation time: 10.9 s - Training time: 101.9 s - Total: 112.8 s\n",
"\n",
"----- Episode 29 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -12264.0 - Epsilon: 0.72\n",
"Training...\n",
"Simulation time: 10.4 s - Training time: 101.5 s - Total: 111.9 s\n",
"\n",
"----- Episode 30 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -14894.0 - Epsilon: 0.71\n",
"Training...\n",
"Simulation time: 10.8 s - Training time: 100.8 s - Total: 111.6 s\n",
"\n",
"----- Episode 31 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
```

```
"Simulating...\n",
"Total reward: -13224.0 - Epsilon: 0.7\n",
"Training...\n",
"Simulation time: 11.0 s - Training time: 101.6 s - Total: 112.6 s\n",
"\n",
"----- Episode 32 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -13646.0 - Epsilon: 0.69\n",
"Training...\n",
"Simulation time: 11.3 s - Training time: 100.9 s - Total: 112.2 s\n",
"\n",
"----- Episode 33 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -13067.0 - Epsilon: 0.68\n",
"Training...\n",
"Simulation time: 11.2 s - Training time: 100.7 s - Total: 111.9 s\n",
"\n",
"----- Episode 34 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
```

"Total reward: -10860.0 - Epsilon: 0.67\n",

"Training...\n",

"Simulation time: 11.4 s - Training time: 101.0 s - Total: 112.4 s\n",

"\n",

"----- Episode 35 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -11620.0 - Epsilon: 0.66\n",

"Training...\n",

"Simulation time: 12.2 s - Training time: 99.6 s - Total: 111.8 s\n",

"\n",

"----- Episode 36 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -12633.0 - Epsilon: 0.65\n",

"Training...\n",

"Simulation time: 13.7 s - Training time: 102.2 s - Total: 115.9 s\n",

"\n",

"----- Episode 37 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -14629.0 - Epsilon: 0.64\n",

```
"Training...\n",
"Simulation time: 11.3 s - Training time: 101.1 s - Total: 112.4 s\n",
"\n",
"----- Episode 38 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -10152.0 - Epsilon: 0.63\n",
"Training...\n",
"Simulation time: 11.5 s - Training time: 101.6 s - Total: 113.1 s\n",
"\n",
"----- Episode 39 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -12381.0 - Epsilon: 0.62\n",
"Training...\n",
"Simulation time: 11.9 s - Training time: 100.8 s - Total: 112.7 s\n",
"\n",
"----- Episode 40 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -10864.0 - Epsilon: 0.61\n",
"Training...\n",
```

```
"Simulation time: 11.8 s - Training time: 100.4 s - Total: 112.2 s\n",
"\n",
"----- Episode 41 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -10807.0 - Epsilon: 0.6\n",
"Training...\n",
"Simulation time: 12.1 s - Training time: 100.5 s - Total: 112.6 s\n",
"\n",
"----- Episode 42 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -13423.0 - Epsilon: 0.59\n",
"Training...\n",
"Simulation time: 12.4 s - Training time: 100.8 s - Total: 113.2 s\n",
"\n",
"----- Episode 43 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -11066.0 - Epsilon: 0.58\n",
"Training...\n",
"Simulation time: 12.0 s - Training time: 100.8 s - Total: 112.8 s\n",
```

"\n",

"----- Episode 44 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -12677.0 - Epsilon: 0.57\n",

"Training...\n",

"Simulation time: 12.1 s - Training time: 102.7 s - Total: 114.8 s\n",

"\n",

"----- Episode 45 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -10007.0 - Epsilon: 0.56\n",

"Training...\n",

"Simulation time: 12.8 s - Training time: 101.6 s - Total: 114.4 s\n",

"\n",

"----- Episode 46 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -14368.0 - Epsilon: 0.55\n",

"Training...\n",

"Simulation time: 12.7 s - Training time: 102.7 s - Total: 115.4 s\n",

"\n",

"----- Episode 47 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -11526.0 - Epsilon: 0.54\n",
"Training...\n",
"Simulation time: 13.8 s - Training time: 100.8 s - Total: 114.6 s\n",
"\n",
"----- Episode 48 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -9760.0 - Epsilon: 0.53\n",
"Training...\n",
"Simulation time: 13.6 s - Training time: 101.8 s - Total: 115.4 s\n",
"\n",
"----- Episode 49 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -9494.0 - Epsilon: 0.52\n",
"Training...\n",
"Simulation time: 12.5 s - Training time: 101.2 s - Total: 113.7 s\n",
"\n",
"----- Episode 50 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -11921.0 - Epsilon: 0.51\n",

"Training...\n",

"Simulation time: 12.7 s - Training time: 101.2 s - Total: 113.9 s\n",

"\n",

"----- Episode 51 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9057.0 - Epsilon: 0.5\n",

"Training...\n",

"Simulation time: 13.6 s - Training time: 105.2 s - Total: 118.8 s\n",

"\n",

"----- Episode 52 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9474.0 - Epsilon: 0.49\n",

"Training...\n",

"Simulation time: 13.4 s - Training time: 102.4 s - Total: 115.8 s\n",

"\n",

"----- Episode 53 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9419.0 - Epsilon: 0.48\n",

"Training...\n",

"Simulation time: 13.5 s - Training time: 103.1 s - Total: 116.6 s\n",

"\n",

"----- Episode 54 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9244.0 - Epsilon: 0.47\n",

"Training...\n",

"Simulation time: 13.3 s - Training time: 100.6 s - Total: 113.9 s\n",

"\n",

"----- Episode 55 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -8648.0 - Epsilon: 0.46\n",

"Training...\n",

"Simulation time: 14.4 s - Training time: 101.6 s - Total: 116.0 s\n",

"\n",

"----- Episode 56 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

```
"Simulating...\n",
"Total reward: -8704.0 - Epsilon: 0.45\n",
"Training...\n",
"Simulation time: 14.1 s - Training time: 104.1 s - Total: 118.2 s\n",
"\n",
"----- Episode 57 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -9178.0 - Epsilon: 0.44\n",
"Training...\n",
"Simulation time: 13.5 s - Training time: 108.3 s - Total: 121.8 s\n",
"\n",
"----- Episode 58 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -9200.0 - Epsilon: 0.43\n",
"Training...\n",
"Simulation time: 15.0 s - Training time: 103.9 s - Total: 118.9 s\n",
"\n",
"----- Episode 59 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
```

"Total reward: -9509.0 - Epsilon: 0.42\n",

"Training...\n",

"Simulation time: 14.1 s - Training time: 102.3 s - Total: 116.4 s\n",

"\n",

"----- Episode 60 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -10851.0 - Epsilon: 0.41\n",

"Training...\n",

"Simulation time: 13.6 s - Training time: 103.3 s - Total: 116.9 s\n",

"\n",

"----- Episode 61 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9615.0 - Epsilon: 0.4\n",

"Training...\n",

"Simulation time: 14.4 s - Training time: 101.9 s - Total: 116.3 s\n",

"\n",

"----- Episode 62 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9582.0 - Epsilon: 0.39\n",

"Training...\n",

"Simulation time: 13.9 s - Training time: 100.1 s - Total: 114.0 s\n",

"\n",

"----- Episode 63 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9316.0 - Epsilon: 0.38\n",

"Training...\n",

"Simulation time: 16.3 s - Training time: 103.6 s - Total: 119.9 s\n",

"\n",

"----- Episode 64 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7602.0 - Epsilon: 0.37\n",

"Training...\n",

"Simulation time: 15.6 s - Training time: 104.2 s - Total: 119.8 s\n",

"\n",

"----- Episode 65 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9040.0 - Epsilon: 0.36\n",

"Training...\n",

"Simulation time: 15.1 s - Training time: 102.4 s - Total: 117.5 s\n",

"\n",

"----- Episode 66 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7611.0 - Epsilon: 0.35\n",

"Training...\n",

"Simulation time: 15.5 s - Training time: 101.5 s - Total: 117.0 s\n",

"\n",

"----- Episode 67 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -6866.0 - Epsilon: 0.34\n",

"Training...\n",

"Simulation time: 15.2 s - Training time: 100.0 s - Total: 115.2 s\n",

"\n",

"----- Episode 68 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7608.0 - Epsilon: 0.33\n",

"Training...\n",

"Simulation time: 14.9 s - Training time: 115.0 s - Total: 129.9 s\n",

```
"\n",
"----- Episode 69 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -6734.0 - Epsilon: 0.32\n",
"Training...\n",
"Simulation time: 17.5 s - Training time: 119.4 s - Total: 136.9 s\n",
"\n",
"----- Episode 70 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -8820.0 - Epsilon: 0.31\n",
"Training...\n",
"Simulation time: 17.4 s - Training time: 122.6 s - Total: 140.0 s\n",
"\n",
"----- Episode 71 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -9551.0 - Epsilon: 0.3\n",
"Training...\n",
"Simulation time: 16.9 s - Training time: 117.8 s - Total: 134.7 s\n",
"\n",
```

"----- Episode 72 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -6653.0 - Epsilon: 0.29\n",

"Training...\n",

"Simulation time: 17.4 s - Training time: 120.7 s - Total: 138.1 s\n",

"\n",

"----- Episode 73 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7386.0 - Epsilon: 0.28\n",

"Training...\n",

"Simulation time: 17.5 s - Training time: 116.9 s - Total: 134.4 s\n",

"\n",

"----- Episode 74 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -8085.0 - Epsilon: 0.27\n",

"Training...\n",

"Simulation time: 17.2 s - Training time: 116.1 s - Total: 133.3 s\n",

"\n",

"----- Episode 75 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7199.0 - Epsilon: 0.26\n",

"Training...\n",

"Simulation time: 17.9 s - Training time: 117.5 s - Total: 135.4 s\n",

"\n",

"----- Episode 76 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7190.0 - Epsilon: 0.25\n",

"Training...\n",

"Simulation time: 18.7 s - Training time: 120.8 s - Total: 139.5 s\n",

"\n",

"----- Episode 77 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7444.0 - Epsilon: 0.24\n",

"Training...\n",

"Simulation time: 18.4 s - Training time: 120.8 s - Total: 139.2 s\n",

"\n",

"----- Episode 78 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -6540.0 - Epsilon: 0.23\n",

"Training...\n",

"Simulation time: 18.0 s - Training time: 118.9 s - Total: 136.9 s\n",

"\n",

"----- Episode 79 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7066.0 - Epsilon: 0.22\n",

"Training...\n",

"Simulation time: 18.2 s - Training time: 121.5 s - Total: 139.7 s\n",

"\n",

"----- Episode 80 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -6798.0 - Epsilon: 0.21\n",

"Training...\n",

"Simulation time: 18.6 s - Training time: 117.2 s - Total: 135.8 s\n",

"\n",

"----- Episode 81 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -6185.0 - Epsilon: 0.2\n",

"Training...\n",

"Simulation time: 18.7 s - Training time: 120.3 s - Total: 139.0 s\n",

"\n",

"----- Episode 82 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -5568.0 - Epsilon: 0.19\n",

"Training...\n",

"Simulation time: 19.6 s - Training time: 116.1 s - Total: 135.7 s\n",

"\n",

"----- Episode 83 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -5111.0 - Epsilon: 0.18\n",

"Training...\n",

"Simulation time: 18.9 s - Training time: 116.9 s - Total: 135.8 s\n",

"\n",

"----- Episode 84 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7017.0 - Epsilon: 0.17\n",

"Training...\n",

"Simulation time: 18.3 s - Training time: 117.1 s - Total: 135.4 s\n",

"\n",

"----- Episode 85 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -4870.0 - Epsilon: 0.16\n",

"Training...\n",

"Simulation time: 19.1 s - Training time: 122.3 s - Total: 141.4 s\n",

"\n",

"----- Episode 86 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -5549.0 - Epsilon: 0.15\n",

"Training...\n",

"Simulation time: 20.3 s - Training time: 124.5 s - Total: 144.8 s\n",

"\n",

"----- Episode 87 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -6581.0 - Epsilon: 0.14\n",

```
"Training...\n",
"Simulation time: 20.4 s - Training time: 123.3 s - Total: 143.7 s\n",
"\n",
"----- Episode 88 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -5874.0 - Epsilon: 0.13\n",
"Training...\n",
"Simulation time: 21.1 s - Training time: 120.7 s - Total: 141.8 s\n",
"\n",
"----- Episode 89 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -5787.0 - Epsilon: 0.12\n",
"Training...\n",
"Simulation time: 20.2 s - Training time: 119.4 s - Total: 139.6 s\n",
"\n",
"----- Episode 90 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -5235.0 - Epsilon: 0.11\n",
"Training...\n",
```

"Simulation time: 20.7 s - Training time: 123.7 s - Total: 144.4 s\n",

"\n",

"----- Episode 91 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -6450.0 - Epsilon: 0.1\n",

"Training...\n",

"Simulation time: 20.2 s - Training time: 124.3 s - Total: 144.5 s\n",

"\n",

"----- Episode 92 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -6450.0 - Epsilon: 0.09\n",

"Training...\n",

"Simulation time: 21.5 s - Training time: 122.9 s - Total: 144.4 s\n",

"\n",

"----- Episode 93 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -4687.0 - Epsilon: 0.08\n",

"Training...\n",

"Simulation time: 23.1 s - Training time: 126.3 s - Total: 149.4 s\n",

"\n",
"----- Episode 94 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -5188.0 - Epsilon: 0.07\n",
"Training...\n",
"Simulation time: 22.6 s - Training time: 120.2 s - Total: 142.8 s\n",
"\n",
"----- Episode 95 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -5608.0 - Epsilon: 0.06\n",
"Training...\n",
"Simulation time: 20.9 s - Training time: 118.5 s - Total: 139.4 s\n",
"\n",
"----- Episode 96 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -5084.0 - Epsilon: 0.05\n",
"Training...\n",
"Simulation time: 19.9 s - Training time: 122.4 s - Total: 142.3 s\n",
"\n",

```
            " Retrying in 1 seconds\n",
      "Loading configuration ... done.\n",
      "Simulating...\n",
      "Total reward: -5022.0 - Epsilon: 0.01\n",
      "Training...\n",
      "Simulation time: 20.9 s - Training time: 115.9 s - Total: 136.8 s\n",
      "\n",
      "----- Start time: 2020-06-23 15:09:58.335455\n",
      "----- End time: 2020-06-23 18:31:18.960007\n",
      "----- Session info saved at: /content/models/model_1/\n",
      "<<<<<<<<<< Completed >>>>>>>>>>\n"
     ],
     "name": "stdout"
    }
   ]
  },
  {
   "cell_type": "markdown",
   "metadata": {
    "id": "EoyQMaWXkyu1"
   },
   "source": [
    "Check the */models/* directory for output files"
   ]
  }
```

APPENDIX 2: Program Code for ANN

```
{
  "nbformat": 4,
  "nbformat_minor": 0,
  "metadata": {
    "anaconda-cloud": {},
    "kernelspec": {
      "name": "arduino"
      "display_name": "arduino"
    },
    "colab": {
      "name": "TLCS.ipynb",
      "provenance": [],
      "collapsed_sections": [],
      "toc_visible": true
    },
    "accelerator": "TPU"
  },
  "cells": [
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "nhde1biSjajW"
      },
      "source": [
```

"# **Setup for TLCS**"
]
},
{
"cell_type": "code",
"metadata": {
"id": "VZZSh853VCpT",
"outputId": "c2d6c77b-3e5f-4c18-eb49-00a8ab491a97",
"colab": {
"base_uri": "https://localhost:8080/",
"height": 51
}
},
"source": [
"!pwd\n",
"!ls\n",
"!rm -r *\n",
"!ls"
],
"execution_count": 1,
"outputs": [
{
"output_type": "stream",
"text": [
"/content\n",

"sample_data\n"
],
"name": "stdout"
}
]
},
{
"cell_type": "code",
"metadata": {
"id": "GA53_oDvE8eN",
"outputId": "6638e416-6098-42ab-d80b-9d8112bf5ac0",
"colab": {
"base_uri": "https://localhost:8080/",
"height": 102
}
},
"source": [
"!git clone \"https://github.com/AndreaVidali/Deep-QLearning-Agent-for-Traffic-Signal-Control\""
],
"execution_count": 2,
"outputs": [
{
"output_type": "stream",
"text": [
"Cloning into 'Deep-QLearning-Agent-for-Traffic-Signal-Control'...\n",

"remote: Enumerating objects: 168, done.\u001b[K\n",
"remote: Total 168 (delta 0), reused 0 (delta 0), pack-reused 168\u001b[K\n",
"Receiving objects: 100% (168/168), 156.58 KiB | 2.06 MiB/s, done.\n",
"Resolving deltas: 100% (81/81), done.\n"
],
"name": "stdout"
}
]
},
{
"cell_type": "code",
"metadata": {
"id": "ZzlPMXYpFXi9"
},
"source": [
"!mv /content/Deep-QLearning-Agent-for-Traffic-Signal-Control/TLCS/* /content/\n",
"!rm -r Deep-QLearning-Agent-for-Traffic-Signal-Control"
],
"execution_count": 3,
"outputs": []
},
{
"cell_type": "code",
"metadata": {
"id": "MVyz7OCYpA85",

"outputId": "1b681078-7a39-4f69-f584-9116da2dacc8",
"colab": {
"base_uri": "https://localhost:8080/",
"height": 238
}
},
"source": [
"!ls -l"
],
"execution_count": 4,
"outputs": [
{
"output_type": "stream",
"text": [
"total 72\n",
"-rw-r--r-- 1 root root 5157 Sep 30 22:26 generator.py\n",
"drwxr-xr-x 2 root root 4096 Sep 30 22:26 intersection\n",
"-rw-r--r-- 1 root root   978 Sep 30 22:26 memory.py\n",
"-rw-r--r-- 1 root root 3312 Sep 30 22:26 model.py\n",
"-rw-r--r-- 1 root root 1752 Sep 30 22:26 testing_main.py\n",
"-rw-r--r-- 1 root root   258 Sep 30 22:26 testing_settings.ini\n",
"-rw-r--r-- 1 root root 8692 Sep 30 22:26 testing_simulation.py\n",
"-rw-r--r-- 1 root root 2794 Sep 30 22:26 training_main.py\n",
"-rw-r--r-- 1 root root   413 Sep 30 22:26 training_settings.ini\n",
"-rw-r--r-- 1 root root 11990 Sep 30 22:26 training_simulation.py\n",

```
      "-rw-r--r-- 1 root root  4658 Sep 30 22:26 utils.py\n",
      "-rw-r--r-- 1 root root  1006 Sep 30 22:26 visualization.py\n"
     ],
     "name": "stdout"
    }
   ]
  },
  {
   "cell_type": "code",
   "metadata": {
    "id": "8zjb7bq_jrSp",
    "outputId": "4143283a-bf3b-4ab2-acb7-6c61aea55d0c",
    "colab": {
     "base_uri": "https://localhost:8080/",
     "height": 34
    }
   },
   "source": [
    "!python --version"
   ],
   "execution_count": null,
   "outputs": [
    {
     "output_type": "stream",
     "text": [
```

"Python 3.6.9\n"
],
"name": "stdout"
}
]
},
{
"cell_type": "markdown",
"metadata": {
"id": "CZ0Ptg7fkXrY"
},
"source": [
"Installing **SUMO TraCI**"
]
},
{
"cell_type": "code",
"metadata": {
"id": "3Oqs5WNyVpco",
"outputId": "d5f5a6cf-da87-4d12-a2c4-0c4d052f0d06",
"colab": {
"base_uri": "https://localhost:8080/",
"height": 1000
}
},

"source": [

"!add-apt-repository ppa:sumo/stable -y\n",

"!apt-get update -y\n",

"!apt-get install sumo sumo-tools sumo-doc"

],

"execution_count": null,

"outputs": [

{

"output_type": "stream",

"text": [

"\r0% [Working]\r \rGet:1 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]\n",

"\r0% [Connecting to archive.ubuntu.com (91.189.88.142)] [1 InRelease 14.2 kB/88.7\r \rGet:2 https://cloud.r-project.org/bin/linux/ubuntu bionic-cran35/ InRelease [3,626 B]\n",

"\r0% [Connecting to archive.ubuntu.com (91.189.88.142)] [1 InRelease 14.2 kB/88.7\r0% [Connecting to archive.ubuntu.com (91.189.88.142)] [1 InRelease 43.1 kB/88.7\r0% [2 InRelease gpgv 3,626 B] [Connecting to archive.ubuntu.com (91.189.88.142)\r \rIgn:3 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 InRelease\n",

"\r0% [2 InRelease gpgv 3,626 B] [Waiting for headers] [1 InRelease 43.1 kB/88.7 k\r0% [2 InRelease gpgv 3,626 B] [Waiting for headers] [Waiting for headers] [Wait\r \rIgn:4 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 InRelease\n",

"Get:5 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 Release [697 B]\n",

"Hit:6 http://archive.ubuntu.com/ubuntu bionic InRelease\n",

"Hit:7 http://ppa.launchpad.net/graphics-drivers/ppa/ubuntu bionic InRelease\n",

"Get:8 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 Release [564 B]\n",

"Get:9 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 Release.gpg [801 B]\n",

"Get:10 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 Release.gpg [833 B]\n",

"Get:11 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]\n",

"Get:12 http://ppa.launchpad.net/marutter/c2d4u3.5/ubuntu bionic InRelease [15.4 kB]\n",

"Get:13 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [961 kB]\n",

"Get:14 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]\n",

"Get:15 http://ppa.launchpad.net/sumo/stable/ubuntu bionic InRelease [15.4 kB]\n",

"Ign:16 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 Packages\n",

"Get:16 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 Packages [151 kB]\n",

"Get:17 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 Packages [38.7 kB]\n",

"Get:18 http://ppa.launchpad.net/marutter/c2d4u3.5/ubuntu bionic/main Sources [1,840 kB]\n",

"Get:19 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [1,397 kB]\n",

"Get:20 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [1,254 kB]\n",

"Get:21 http://ppa.launchpad.net/marutter/c2d4u3.5/ubuntu bionic/main amd64 Packages [888 kB]\n",

"Get:22 http://ppa.launchpad.net/sumo/stable/ubuntu bionic/main amd64 Packages [901 B]\n",

"Fetched 6,819 kB in 4s (1,813 kB/s)\n",

"Reading package lists... Done\n",

"Hit:1 https://cloud.r-project.org/bin/linux/ubuntu bionic-cran35/ InRelease\n",

"Hit:2 http://security.ubuntu.com/ubuntu bionic-security InRelease\n",

"Ign:3 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 InRelease\n", "Ign:4 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 InRelease\n",

"Hit:5 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 Release\n",

"Hit:6 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 Release\n",

"Hit:7 http://archive.ubuntu.com/ubuntu bionic InRelease\n",

"Hit:8 http://ppa.launchpad.net/graphics-drivers/ppa/ubuntu bionic InRelease\n",

"Hit:9 http://archive.ubuntu.com/ubuntu bionic-updates InRelease\n",

"Hit:10 http://archive.ubuntu.com/ubuntu bionic-backports InRelease\n",

"Hit:11 http://ppa.launchpad.net/marutter/c2d4u3.5/ubuntu bionic InRelease\n",

"Hit:12 http://ppa.launchpad.net/sumo/stable/ubuntu bionic InRelease\n",

"Reading package lists... Done\n",

"Reading package lists... Done\n",

"Building dependency tree   \n",

"Reading state information... Done\n",

"The following package was automatically installed and is no longer required:\n",

" libnvidia-common-440\n",

"Use 'apt autoremove' to remove it.\n",

"The following additional packages will be installed:\n",

" binfmt-support fastjar jarwrapper javascript-common libfox-1.6-0\n",

" libjs-jquery libjs-openlayers libjs-underscore\n",

"Suggested packages:\n",

" apache2 | lighttpd | httpd\n",

"The following NEW packages will be installed:\n",

" binfmt-support fastjar jarwrapper javascript-common libfox-1.6-0\n",

" libjs-jquery libjs-openlayers libjs-underscore sumo sumo-doc sumo-tools\n", "0 upgraded, 11 newly installed, 0 to remove and 53 not upgraded.\n",

"Need to get 25.6 MB of archives.\n",

"After this operation, 154 MB of additional disk space will be used.\n",

"Get:1 http://archive.ubuntu.com/ubuntu bionic/main amd64 binfmt-support amd64 2.1.8-2 [51.6 kB]\n",

"Get:2 http://ppa.launchpad.net/sumo/stable/ubuntu bionic/main amd64 sumo amd64 1.6.0+dfsg1-1 [12.0 MB]\n",

"Get:3 http://archive.ubuntu.com/ubuntu bionic/universe amd64 fastjar amd64 2:0.98-6build1 [66.7 kB]\n",

"Get:4 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 jarwrapper all 0.72.1~18.04.1 [20.3 kB]\n",

"Get:5 http://archive.ubuntu.com/ubuntu bionic/main amd64 javascript-common all 11 [6,066 B]\n",

"Get:6 http://archive.ubuntu.com/ubuntu bionic/universe amd64 libfox-1.6-0 amd64 1.6.56-1 [875 kB]\n",

"Get:7 http://archive.ubuntu.com/ubuntu bionic/main amd64 libjs-jquery all 3.2.1-1 [152 kB]\n",

"Get:8 http://archive.ubuntu.com/ubuntu bionic/main amd64 libjs-underscore all 1.8.3~dfsg-1 [59.9 kB]\n",

"Get:9 http://archive.ubuntu.com/ubuntu bionic/universe amd64 libjs-openlayers all 2.13.1+ds2-4 [704 kB]\n",

"Get:10 http://ppa.launchpad.net/sumo/stable/ubuntu bionic/main amd64 sumo-doc all 1.6.0+dfsg1-1 [787 kB]\n",

"Get:11 http://ppa.launchpad.net/sumo/stable/ubuntu bionic/main amd64 sumo-tools all 1.6.0+dfsg1-1 [10.9 MB]\n",

"Fetched 25.6 MB in 4s (6,822 kB/s)\n",

"Selecting previously unselected package binfmt-support.\n",

"(Reading database ... 144328 files and directories currently installed.)\n",

"Preparing to unpack .../00-binfmt-support_2.1.8-2_amd64.deb ...\n",

"Unpacking binfmt-support (2.1.8-2) ...\n",

"Selecting previously unselected package fastjar.\n",

"Preparing to unpack .../01-fastjar_2%3a0.98-6build1_amd64.deb ...\n",

"Unpacking fastjar (2:0.98-6build1) ...\n",

"Selecting previously unselected package jarwrapper.\n",

"Preparing to unpack .../02-jarwrapper_0.72.1~18.04.1_all.deb ...\n",

"Unpacking jarwrapper (0.72.1~18.04.1) ...\n",

"Selecting previously unselected package javascript-common.\n",

"Preparing to unpack .../03-javascript-common_11_all.deb ...\n",

"Unpacking javascript-common (11) ...\n",

"Selecting previously unselected package libfox-1.6-0:amd64.\n",

"Preparing to unpack .../04-libfox-1.6-0_1.6.56-1_amd64.deb ...\n",

"Unpacking libfox-1.6-0:amd64 (1.6.56-1) ...\n",

"Selecting previously unselected package libjs-jquery.\n",

"Preparing to unpack .../05-libjs-jquery_3.2.1-1_all.deb ...\n",

"Unpacking libjs-jquery (3.2.1-1) ...\n",

"Selecting previously unselected package libjs-underscore.\n",

"Preparing to unpack .../06-libjs-underscore_1.8.3~dfsg-1_all.deb ...\n",

"Unpacking libjs-underscore (1.8.3~dfsg-1) ...\n",

"Selecting previously unselected package sumo.\n",

"Preparing to unpack .../07-sumo_1.6.0+dfsg1-1_amd64.deb ...\n",

"Unpacking sumo (1.6.0+dfsg1-1) ...\n",

"Selecting previously unselected package sumo-doc.\n",

"Preparing to unpack .../08-sumo-doc_1.6.0+dfsg1-1_all.deb ...\n",

"Unpacking sumo-doc (1.6.0+dfsg1-1) ...\n",

"Selecting previously unselected package libjs-openlayers.\n",

"Preparing to unpack .../09-libjs-openlayers_2.13.1+ds2-4_all.deb ...\n",

"Unpacking libjs-openlayers (2.13.1+ds2-4) ...\n",

"Selecting previously unselected package sumo-tools.\n",

"Preparing to unpack .../10-sumo-tools_1.6.0+dfsg1-1_all.deb ...\n",

"Unpacking sumo-tools (1.6.0+dfsg1-1) ...\n",

"Setting up libjs-jquery (3.2.1-1) ...\n",

"Setting up binfmt-support (2.1.8-2) ...\n",

"Created symlink /etc/systemd/system/multi-user.target.wants/binfmt-support.service ? /lib/systemd/system/binfmt-support.service.\n",

"invoke-rc.d: could not determine current runlevel\n",

"invoke-rc.d: policy-rc.d denied execution of start.\n",

"Setting up libfox-1.6-0:amd64 (1.6.56-1) ...\n",

"Setting up libjs-underscore (1.8.3~dfsg-1) ...\n",

"Setting up fastjar (2:0.98-6build1) ...\n",

"Setting up libjs-openlayers (2.13.1+ds2-4) ...\n",

"Setting up sumo (1.6.0+dfsg1-1) ...\n",

"Setting up javascript-common (11) ...\n",

"Setting up jarwrapper (0.72.1~18.04.1) ...\n",

"Setting up sumo-doc (1.6.0+dfsg1-1) ...\n",

"Setting up sumo-tools (1.6.0+dfsg1-1) ...\n",

"Processing triggers for systemd (237-3ubuntu10.41) ...\n",

"Processing triggers for man-db (2.8.3-2ubuntu0.1) ...\n",

"Processing triggers for mime-support (3.60ubuntu1) ...\n",

"Processing triggers for libc-bin (2.27-3ubuntu1) ...\n",

"/sbin/ldconfig.real: /usr/local/lib/python3.6/dist-packages/ideep4py/lib/libmkldnn.so.0 is not a symbolic link\n",

```
        "\n"
      ],
      "name": "stdout"
    }
  ]
},
{
  "cell_type": "code",
  "metadata": {
    "id": "AkoK63N3mR4E",
    "outputId": "f0efee91-beb1-48cc-b1db-e502d07d27c0",
    "colab": {
      "base_uri": "https://localhost:8080/",
      "height": 173
    }
  },
  "source": [
    "!pip install traci"
  ],
  "execution_count": null,
  "outputs": [
    {
      "output_type": "stream",
      "text": [
        "Collecting traci\n",
```

"\u001b[?25l Downloading https://files.pythonhosted.org/packages/e6/93/3c68e9f61c8415bf727ae3ac48627c54e4141d284ba0b37c5da810939ca6/traci-1.6.0-py3-none-any.whl (206kB)\n",

"\u001b[K | | 215kB 3.5MB/s \n",

"\u001b[?25hCollecting sumolib>=1.6.0\n",

"\u001b[?25l Downloading https://files.pythonhosted.org/packages/67/f2/c026667c006420cf98948ec3a81f4a68da2f386e14c323ad917462a74267/sumolib-1.6.0-py3-none-any.whl (130kB)\n",

"\u001b[K | | 133kB 10.6MB/s \n",

"\u001b[?25hInstalling collected packages: sumolib, traci\n",

"Successfully installed sumolib-1.6.0 traci-1.6.0\n"

],

"name": "stdout"

}

]

},

{

"cell_type": "code",

"metadata": {

"id": "s-6v3Jk1oxwE"

},

"source": [

"import os\n",

"os.environ['SUMO_HOME'] = \"/usr/share/sumo/\""

],

"execution_count": null,

"outputs": []

},
{
"cell_type": "markdown",
"metadata": {
"id": "ebzkYv4_klSV"
},
"source": [
"**Starting Training Process**"
]
},
{
"cell_type": "code",
"metadata": {
"id": "b2ACqSSajWoT",
"outputId": "031b34d7-bca5-4eba-8700-3ff292fd9c92",
"colab": {
"base_uri": "https://localhost:8080/",
"height": 1000
}
},
"source": [
"!python training_main.py"
],
"execution_count": null,
"outputs": [

{

"output_type": "stream",

"text": [

"2020-06-23 15:09:58.148129: E tensorflow/stream_executor/cuda/cuda_driver.cc:313] failed call to cuInit: CUDA_ERROR_NO_DEVICE: no CUDA-capable device is detected\n",

"\n",

"----- Episode 1 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -24710.0 - Epsilon: 1.0\n",

"Training...\n",

"Simulation time: 8.4 s - Training time: 0.0 s - Total: 8.4 s\n",

"\n",

"----- Episode 2 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -26416.0 - Epsilon: 0.99\n",

"Training...\n",

"Simulation time: 8.2 s - Training time: 101.4 s - Total: 109.6 s\n",

"\n",

"----- Episode 3 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -35112.0 - Epsilon: 0.98\n",

"Training...\n",

"Simulation time: 8.6 s - Training time: 102.2 s - Total: 110.8 s\n",

"\n",

"----- Episode 4 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -31930.0 - Epsilon: 0.97\n",

"Training...\n",

"Simulation time: 8.7 s - Training time: 100.4 s - Total: 109.1 s\n",

"\n",

"----- Episode 5 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -22242.0 - Epsilon: 0.96\n",

"Training...\n",

"Simulation time: 8.4 s - Training time: 102.8 s - Total: 111.2 s\n",

"\n",

"----- Episode 6 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -34009.0 - Epsilon: 0.95\n",

"Training...\n",

"Simulation time: 9.4 s - Training time: 105.9 s - Total: 115.3 s\n",

"\n",

"----- Episode 7 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -17112.0 - Epsilon: 0.94\n",

"Training...\n",

"Simulation time: 8.4 s - Training time: 102.4 s - Total: 110.8 s\n",

"\n",

"----- Episode 8 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -20555.0 - Epsilon: 0.93\n",

"Training...\n",

"Simulation time: 9.1 s - Training time: 101.4 s - Total: 110.5 s\n",

"\n",

"----- Episode 9 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -19614.0 - Epsilon: 0.92\n",

"Training...\n",

```
"Simulation time: 8.8 s - Training time: 100.6 s - Total: 109.4 s\n",
"\n",
"----- Episode 10 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -23736.0 - Epsilon: 0.91\n",
"Training...\n",
"Simulation time: 8.8 s - Training time: 99.3 s - Total: 108.1 s\n",
"\n",
"----- Episode 11 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -24066.0 - Epsilon: 0.9\n",
"Training...\n",
"Simulation time: 9.2 s - Training time: 99.2 s - Total: 108.4 s\n",
"\n",
"----- Episode 12 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -25121.0 - Epsilon: 0.89\n",
"Training...\n",
"Simulation time: 9.0 s - Training time: 100.2 s - Total: 109.2 s\n",
```

"\n",

"----- Episode 13 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -16412.0 - Epsilon: 0.88\n",

"Training...\n",

"Simulation time: 9.7 s - Training time: 99.9 s - Total: 109.6 s\n",

"\n",

"----- Episode 14 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -19786.0 - Epsilon: 0.87\n",

"Training...\n",

"Simulation time: 9.0 s - Training time: 100.6 s - Total: 109.6 s\n",

"\n",

"----- Episode 15 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -16815.0 - Epsilon: 0.86\n",

"Training...\n",

"Simulation time: 9.1 s - Training time: 99.7 s - Total: 108.8 s\n",

"\n",

```
"----- Episode 16 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -30985.0 - Epsilon: 0.85\n",
"Training...\n",
"Simulation time: 9.7 s - Training time: 100.7 s - Total: 110.4 s\n",
"\n",
"----- Episode 17 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -18319.0 - Epsilon: 0.84\n",
"Training...\n",
"Simulation time: 9.3 s - Training time: 100.5 s - Total: 109.8 s\n",
"\n",
"----- Episode 18 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -19557.0 - Epsilon: 0.83\n",
"Training...\n",
"Simulation time: 9.4 s - Training time: 101.0 s - Total: 110.4 s\n",
"\n",
"----- Episode 19 of 100\n",
```

```
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -18599.0 - Epsilon: 0.82\n",
"Training...\n",
"Simulation time: 9.4 s - Training time: 100.4 s - Total: 109.8 s\n",
"\n",
"----- Episode 20 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -15339.0 - Epsilon: 0.81\n",
"Training...\n",
"Simulation time: 9.8 s - Training time: 100.2 s - Total: 110.0 s\n",
"\n",
"----- Episode 21 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -20444.0 - Epsilon: 0.8\n",
"Training...\n",
"Simulation time: 10.2 s - Training time: 99.8 s - Total: 110.0 s\n",
"\n",
"----- Episode 22 of 100\n",
" Retrying in 1 seconds\n",
```

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -16207.0 - Epsilon: 0.79\n",

"Training...\n",

"Simulation time: 10.4 s - Training time: 101.3 s - Total: 111.7 s\n",

"\n",

"----- Episode 23 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Warning: Choosing new speed factor 1.25 for vehicle 'N_E_255' to match departure speed.\n",

"Total reward: -15486.0 - Epsilon: 0.78\n",

"Training...\n",

"Simulation time: 10.0 s - Training time: 100.0 s - Total: 110.0 s\n",

"\n",

"----- Episode 24 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -14845.0 - Epsilon: 0.77\n",

"Training...\n",

"Simulation time: 10.6 s - Training time: 101.0 s - Total: 111.6 s\n",

"\n",

"----- Episode 25 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -15499.0 - Epsilon: 0.76\n",

"Training...\n",

"Simulation time: 10.8 s - Training time: 101.7 s - Total: 112.5 s\n",

"\n",

"----- Episode 26 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -14869.0 - Epsilon: 0.75\n",

"Training...\n",

"Simulation time: 10.8 s - Training time: 103.6 s - Total: 114.4 s\n",

"\n",

"----- Episode 27 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -13010.0 - Epsilon: 0.74\n",

"Training...\n",

"Simulation time: 10.9 s - Training time: 102.6 s - Total: 113.5 s\n",

"\n",

"----- Episode 28 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -16417.0 - Epsilon: 0.73\n",

"Training...\n",

"Simulation time: 10.9 s - Training time: 101.9 s - Total: 112.8 s\n",

"\n",

"----- Episode 29 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -12264.0 - Epsilon: 0.72\n",

"Training...\n",

"Simulation time: 10.4 s - Training time: 101.5 s - Total: 111.9 s\n",

"\n",

"----- Episode 30 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -14894.0 - Epsilon: 0.71\n",

"Training...\n",

"Simulation time: 10.8 s - Training time: 100.8 s - Total: 111.6 s\n",

"\n",

"----- Episode 31 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -13224.0 - Epsilon: 0.7\n",

"Training...\n",

"Simulation time: 11.0 s - Training time: 101.6 s - Total: 112.6 s\n",

"\n",

"----- Episode 32 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -13646.0 - Epsilon: 0.69\n",

"Training...\n",

"Simulation time: 11.3 s - Training time: 100.9 s - Total: 112.2 s\n",

"\n",

"----- Episode 33 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -13067.0 - Epsilon: 0.68\n",

"Training...\n",

"Simulation time: 11.2 s - Training time: 100.7 s - Total: 111.9 s\n",

"\n",

"----- Episode 34 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -10860.0 - Epsilon: 0.67\n",

"Training...\n",

"Simulation time: 11.4 s - Training time: 101.0 s - Total: 112.4 s\n",

"\n",

"----- Episode 35 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -11620.0 - Epsilon: 0.66\n",

"Training...\n",

"Simulation time: 12.2 s - Training time: 99.6 s - Total: 111.8 s\n",

"\n",

"----- Episode 36 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -12633.0 - Epsilon: 0.65\n",

"Training...\n",

"Simulation time: 13.7 s - Training time: 102.2 s - Total: 115.9 s\n",

"\n",

"----- Episode 37 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -14629.0 - Epsilon: 0.64\n",

"Training...\n",

"Simulation time: 11.3 s - Training time: 101.1 s - Total: 112.4 s\n",

"\n",

"----- Episode 38 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -10152.0 - Epsilon: 0.63\n",

"Training...\n",

"Simulation time: 11.5 s - Training time: 101.6 s - Total: 113.1 s\n",

"\n",

"----- Episode 39 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -12381.0 - Epsilon: 0.62\n",

"Training...\n",

"Simulation time: 11.9 s - Training time: 100.8 s - Total: 112.7 s\n",

"\n",

"----- Episode 40 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -10864.0 - Epsilon: 0.61\n",

"Training...\n",

"Simulation time: 11.8 s - Training time: 100.4 s - Total: 112.2 s\n",

"\n",
"----- Episode 41 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -10807.0 - Epsilon: 0.6\n",
"Training...\n",
"Simulation time: 12.1 s - Training time: 100.5 s - Total: 112.6 s\n",
"\n",
"----- Episode 42 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -13423.0 - Epsilon: 0.59\n",
"Training...\n",
"Simulation time: 12.4 s - Training time: 100.8 s - Total: 113.2 s\n",
"\n",
"----- Episode 43 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -11066.0 - Epsilon: 0.58\n",
"Training...\n",
"Simulation time: 12.0 s - Training time: 100.8 s - Total: 112.8 s\n",
"\n",

```
"----- Episode 44 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -12677.0 - Epsilon: 0.57\n",
"Training...\n",
"Simulation time: 12.1 s - Training time: 102.7 s - Total: 114.8 s\n",
"\n",
"----- Episode 45 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -10007.0 - Epsilon: 0.56\n",
"Training...\n",
"Simulation time: 12.8 s - Training time: 101.6 s - Total: 114.4 s\n",
"\n",
"----- Episode 46 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -14368.0 - Epsilon: 0.55\n",
"Training...\n",
"Simulation time: 12.7 s - Training time: 102.7 s - Total: 115.4 s\n",
"\n",
"----- Episode 47 of 100\n",
```

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -11526.0 - Epsilon: 0.54\n",

"Training...\n",

"Simulation time: 13.8 s - Training time: 100.8 s - Total: 114.6 s\n",

"\n",

"----- Episode 48 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9760.0 - Epsilon: 0.53\n",

"Training...\n",

"Simulation time: 13.6 s - Training time: 101.8 s - Total: 115.4 s\n",

"\n",

"----- Episode 49 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9494.0 - Epsilon: 0.52\n",

"Training...\n",

"Simulation time: 12.5 s - Training time: 101.2 s - Total: 113.7 s\n",

"\n",

"----- Episode 50 of 100\n",

" Retrying in 1 seconds\n",

```
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -11921.0 - Epsilon: 0.51\n",
"Training...\n",
"Simulation time: 12.7 s - Training time: 101.2 s - Total: 113.9 s\n",
"\n",
"----- Episode 51 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -9057.0 - Epsilon: 0.5\n",
"Training...\n",
"Simulation time: 13.6 s - Training time: 105.2 s - Total: 118.8 s\n",
"\n",
"----- Episode 52 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -9474.0 - Epsilon: 0.49\n",
"Training...\n",
"Simulation time: 13.4 s - Training time: 102.4 s - Total: 115.8 s\n",
"\n",
"----- Episode 53 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
```

"Simulating...\n",

"Total reward: -9419.0 - Epsilon: 0.48\n",

"Training...\n",

"Simulation time: 13.5 s - Training time: 103.1 s - Total: 116.6 s\n",

"\n",

"----- Episode 54 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9244.0 - Epsilon: 0.47\n",

"Training...\n",

"Simulation time: 13.3 s - Training time: 100.6 s - Total: 113.9 s\n",

"\n",

"----- Episode 55 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -8648.0 - Epsilon: 0.46\n",

"Training...\n",

"Simulation time: 14.4 s - Training time: 101.6 s - Total: 116.0 s\n",

"\n",

"----- Episode 56 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -8704.0 - Epsilon: 0.45\n",

"Training...\n",

"Simulation time: 14.1 s - Training time: 104.1 s - Total: 118.2 s\n",

"\n",

"----- Episode 57 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9178.0 - Epsilon: 0.44\n",

"Training...\n",

"Simulation time: 13.5 s - Training time: 108.3 s - Total: 121.8 s\n",

"\n",

"----- Episode 58 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9200.0 - Epsilon: 0.43\n",

"Training...\n",

"Simulation time: 15.0 s - Training time: 103.9 s - Total: 118.9 s\n",

"\n",

"----- Episode 59 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9509.0 - Epsilon: 0.42\n",

```
"Training...\n",
"Simulation time: 14.1 s - Training time: 102.3 s - Total: 116.4 s\n",
"\n",
"----- Episode 60 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -10851.0 - Epsilon: 0.41\n",
"Training...\n",
"Simulation time: 13.6 s - Training time: 103.3 s - Total: 116.9 s\n",
"\n",
"----- Episode 61 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -9615.0 - Epsilon: 0.4\n",
"Training...\n",
"Simulation time: 14.4 s - Training time: 101.9 s - Total: 116.3 s\n",
"\n",
"----- Episode 62 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -9582.0 - Epsilon: 0.39\n",
"Training...\n",
```

```
"Simulation time: 13.9 s - Training time: 100.1 s - Total: 114.0 s\n",

"\n",

"----- Episode 63 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9316.0 - Epsilon: 0.38\n",

"Training...\n",

"Simulation time: 16.3 s - Training time: 103.6 s - Total: 119.9 s\n",

"\n",

"----- Episode 64 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7602.0 - Epsilon: 0.37\n",

"Training...\n",

"Simulation time: 15.6 s - Training time: 104.2 s - Total: 119.8 s\n",

"\n",

"----- Episode 65 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9040.0 - Epsilon: 0.36\n",

"Training...\n",

"Simulation time: 15.1 s - Training time: 102.4 s - Total: 117.5 s\n",
```

"\n",

"----- Episode 66 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7611.0 - Epsilon: 0.35\n",

"Training...\n",

"Simulation time: 15.5 s - Training time: 101.5 s - Total: 117.0 s\n",

"\n",

"----- Episode 67 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -6866.0 - Epsilon: 0.34\n",

"Training...\n",

"Simulation time: 15.2 s - Training time: 100.0 s - Total: 115.2 s\n",

"\n",

"----- Episode 68 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7608.0 - Epsilon: 0.33\n",

"Training...\n",

"Simulation time: 14.9 s - Training time: 115.0 s - Total: 129.9 s\n",

"\n",

"----- Episode 69 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -6734.0 - Epsilon: 0.32\n",

"Training...\n",

"Simulation time: 17.5 s - Training time: 119.4 s - Total: 136.9 s\n",

"\n",

"----- Episode 70 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -8820.0 - Epsilon: 0.31\n",

"Training...\n",

"Simulation time: 17.4 s - Training time: 122.6 s - Total: 140.0 s\n",

"\n",

"----- Episode 71 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -9551.0 - Epsilon: 0.3\n",

"Training...\n",

"Simulation time: 16.9 s - Training time: 117.8 s - Total: 134.7 s\n",

"\n",

"----- Episode 72 of 100\n",

```
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -6653.0 - Epsilon: 0.29\n",
"Training...\n",
"Simulation time: 17.4 s - Training time: 120.7 s - Total: 138.1 s\n",
"\n",
"----- Episode 73 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -7386.0 - Epsilon: 0.28\n",
"Training...\n",
"Simulation time: 17.5 s - Training time: 116.9 s - Total: 134.4 s\n",
"\n",
"----- Episode 74 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -8085.0 - Epsilon: 0.27\n",
"Training...\n",
"Simulation time: 17.2 s - Training time: 116.1 s - Total: 133.3 s\n",
"\n",
"----- Episode 75 of 100\n",
" Retrying in 1 seconds\n",
```

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7199.0 - Epsilon: 0.26\n",

"Training...\n",

"Simulation time: 17.9 s - Training time: 117.5 s - Total: 135.4 s\n",

"\n",

"----- Episode 76 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7190.0 - Epsilon: 0.25\n",

"Training...\n",

"Simulation time: 18.7 s - Training time: 120.8 s - Total: 139.5 s\n",

"\n",

"----- Episode 77 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7444.0 - Epsilon: 0.24\n",

"Training...\n",

"Simulation time: 18.4 s - Training time: 120.8 s - Total: 139.2 s\n",

"\n",

"----- Episode 78 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

```
"Simulating...\n",
"Total reward: -6540.0 - Epsilon: 0.23\n",
"Training...\n",
"Simulation time: 18.0 s - Training time: 118.9 s - Total: 136.9 s\n",
"\n",
"----- Episode 79 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -7066.0 - Epsilon: 0.22\n",
"Training...\n",
"Simulation time: 18.2 s - Training time: 121.5 s - Total: 139.7 s\n",
"\n",
"----- Episode 80 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -6798.0 - Epsilon: 0.21\n",
"Training...\n",
"Simulation time: 18.6 s - Training time: 117.2 s - Total: 135.8 s\n",
"\n",
"----- Episode 81 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
```

"Total reward: -6185.0 - Epsilon: 0.2\n",

"Training...\n",

"Simulation time: 18.7 s - Training time: 120.3 s - Total: 139.0 s\n",

"\n",

"----- Episode 82 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -5568.0 - Epsilon: 0.19\n",

"Training...\n",

"Simulation time: 19.6 s - Training time: 116.1 s - Total: 135.7 s\n",

"\n",

"----- Episode 83 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -5111.0 - Epsilon: 0.18\n",

"Training...\n",

"Simulation time: 18.9 s - Training time: 116.9 s - Total: 135.8 s\n",

"\n",

"----- Episode 84 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -7017.0 - Epsilon: 0.17\n",

"Training...\n",

"Simulation time: 18.3 s - Training time: 117.1 s - Total: 135.4 s\n",

"\n",

"----- Episode 85 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -4870.0 - Epsilon: 0.16\n",

"Training...\n",

"Simulation time: 19.1 s - Training time: 122.3 s - Total: 141.4 s\n",

"\n",

"----- Episode 86 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -5549.0 - Epsilon: 0.15\n",

"Training...\n",

"Simulation time: 20.3 s - Training time: 124.5 s - Total: 144.8 s\n",

"\n",

"----- Episode 87 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -6581.0 - Epsilon: 0.14\n",

"Training...\n",

"Simulation time: 20.4 s - Training time: 123.3 s - Total: 143.7 s\n",

"\n",

"----- Episode 88 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -5874.0 - Epsilon: 0.13\n",

"Training...\n",

"Simulation time: 21.1 s - Training time: 120.7 s - Total: 141.8 s\n",

"\n",

"----- Episode 89 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -5787.0 - Epsilon: 0.12\n",

"Training...\n",

"Simulation time: 20.2 s - Training time: 119.4 s - Total: 139.6 s\n",

"\n",

"----- Episode 90 of 100\n",

" Retrying in 1 seconds\n",

"Loading configuration ... done.\n",

"Simulating...\n",

"Total reward: -5235.0 - Epsilon: 0.11\n",

"Training...\n",

"Simulation time: 20.7 s - Training time: 123.7 s - Total: 144.4 s\n",

```
"\n",
"----- Episode 91 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -6450.0 - Epsilon: 0.1\n",
"Training...\n",
"Simulation time: 20.2 s - Training time: 124.3 s - Total: 144.5 s\n",
"\n",
"----- Episode 92 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -6450.0 - Epsilon: 0.09\n",
"Training...\n",
"Simulation time: 21.5 s - Training time: 122.9 s - Total: 144.4 s\n",
"\n",
"----- Episode 93 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -4687.0 - Epsilon: 0.08\n",
"Training...\n",
"Simulation time: 23.1 s - Training time: 126.3 s - Total: 149.4 s\n",
"\n",
```

```
"----- Episode 94 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -5188.0 - Epsilon: 0.07\n",
"Training...\n",
"Simulation time: 22.6 s - Training time: 120.2 s - Total: 142.8 s\n",
"\n",
"----- Episode 95 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -5608.0 - Epsilon: 0.06\n",
"Training...\n",
"Simulation time: 20.9 s - Training time: 118.5 s - Total: 139.4 s\n",
"\n",
"----- Episode 96 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -5084.0 - Epsilon: 0.05\n",
"Training...\n",
"Simulation time: 19.9 s - Training time: 122.4 s - Total: 142.3 s\n",
"\n",
"----- Episode 97 of 100\n",
```

```
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -5318.0 - Epsilon: 0.04\n",
"Training...\n",
"Simulation time: 20.9 s - Training time: 120.0 s - Total: 140.9 s\n",
"\n",
"----- Episode 98 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -4610.0 - Epsilon: 0.03\n",
"Training...\n",
"Simulation time: 21.0 s - Training time: 121.5 s - Total: 142.5 s\n",
"\n",
"----- Episode 99 of 100\n",
" Retrying in 1 seconds\n",
"Loading configuration ... done.\n",
"Simulating...\n",
"Total reward: -4863.0 - Epsilon: 0.02\n",
"Training...\n",
"Simulation time: 21.6 s - Training time: 120.2 s - Total: 141.8 s\n",
"\n",
"----- Episode 100 of 100\n",
" Retrying in 1 seconds\n",
```

```
      "Loading configuration ... done.\n",
      "Simulating...\n",
      "Total reward: -5022.0 - Epsilon: 0.01\n",
      "Training...\n",
      "Simulation time: 20.9 s - Training time: 115.9 s - Total: 136.8 s\n",
      "\n",
      "----- Start time: 2020-06-23 15:09:58.335455\n",
      "----- End time: 2020-06-23 18:31:18.960007\n",
      "----- Session info saved at: /content/models/model_1/\n",
      "<<<<<<<<<< Completed >>>>>>>>>>\n"
     ],
     "name": "stdout"
    }
   ]
  },
  {
   "cell_type": "markdown",
   "metadata": {
    "id": "EoyQMaWXkyu1"
   },
   "source": [
    "Check the */models/* directory for output files"
   ]
  }
```

APPENDIX 3: SMS Gateway

```
require('babel-register');
require('babel-polyfill');
module.exports = {
  networks: {
    development: {
      host: "127.0.0.1",          //The IP Address
      port: 7545,                 //The localhost port
    },
  },
  compilers: {
    solc: {
      optimizer: {
        enabled: true,
        runs: 200
      }
    }
  }
}

//JavaScript Background Service Worker
const isLocalhost = Boolean(
  window.location.hostname === 'localhost' ||
    // [::1] is the IPv6 localhost address.
```

```
    window.location.hostname === '[::1]' ||
    // 127.0.0.1/8 is considered localhost for IPv4.
    window.location.hostname.match(
      /^127(?:\.(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)){3}$/
    )
);

export function register(config) {
  if (process.env.NODE_ENV === 'production' && 'serviceWorker' in navigator) {
    // The URL constructor is available in all browsers that support SW.
    const publicUrl = new URL(process.env.PUBLIC_URL, window.location.href);
    if (publicUrl.origin !== window.location.origin) {
      // Our service worker won't work if PUBLIC_URL is on a different origin
      // from what our page is served on. This might happen if a CDN is used to
      // serve assets; see https://github.com/facebook/create-react-app/issues/2374
      return;
    }

    window.addEventListener('load', () => {
      const swUrl = `${process.env.PUBLIC_URL}/service-worker.js`;

      if (isLocalhost) {
        // This is running on localhost. Let's check if a service worker still exists or not.
        checkValidServiceWorker(swUrl, config);
```

```
        // Add some additional logging to localhost, pointing developers to the
        // service worker/PWA documentation.
        navigator.serviceWorker.ready.then(() => {
          console.log(
            'This web app is being served cache-first by a service ' +
              'worker. To learn more, visit https://bit.ly/CRA-PWA'
          );
        });
      } else {
        // Is not localhost. Just register service worker
        registerValidSW(swUrl, config);
      }
    });
  }
}

function registerValidSW(swUrl, config) {
  navigator.serviceWorker
    .register(swUrl)
    .then(registration => {
      registration.onupdatefound = () => {
        const installingWorker = registration.installing;
        if (installingWorker == null) {
```

```
  return;
}
installingWorker.onstatechange = () => {
  if (installingWorker.state === 'installed') {
    if (navigator.serviceWorker.controller) {
      // At this point, the updated precached content has been fetched,
      // but the previous service worker will still serve the older
      // content until all client tabs are closed.
      console.log(
        'New content is available and will be used when all ' +
          'tabs for this page are closed. See https://bit.ly/CRA-PWA.'
      );

      // Execute callback
      if (config && config.onUpdate) {
        config.onUpdate(registration);
      }
    } else {
      // At this point, everything has been precached.
      // It's the perfect time to display a
      // "Content is cached for offline use." message.
      console.log('Content is cached for offline use.');

      // Execute callback
```

```
            if (config && config.onSuccess) {
              config.onSuccess(registration);
            }
          }
        }
      };
    };
  })
  .catch(error => {
    console.error('Error during service worker registration:', error);
  });
}

function checkValidServiceWorker(swUrl, config) {
  // Check if the service worker can be found. If it can't reload the page.
  fetch(swUrl)
    .then(response => {
      // Ensure service worker exists, and that we really are getting a JS file.
      const contentType = response.headers.get('content-type');
      if (
        response.status === 404 ||
        (contentType != null && contentType.indexOf('javascript') === -1)
      ) {
        // No service worker found. Probably a different app. Reload the page.
```

```
        navigator.serviceWorker.ready.then(registration => {
          registration.unregister().then(() => {
            window.location.reload();
          });
        });
      } else {
        // Service worker found. Proceed as normal.
        registerValidSW(swUrl, config);
      }
    })
    .catch(() => {
      console.log(
        'No internet connection found. App is running in offline mode.'
      );
    });
}

export function unregister() {
  if ('serviceWorker' in navigator) {
    navigator.serviceWorker.ready.then(registration => {
      registration.unregister();
    });
  }
}
```

```
"dependencies": {
  "babel-polyfill": "6.26.0",
  "babel-preset-env": "1.7.0",
  "babel-preset-es2015": "6.24.1",
  "babel-preset-stage-2": "6.24.1",
  "babel-preset-stage-3": "6.24.1",
  "babel-register": "6.26.0",
  "bootstrap": "4.3.1",
  "chai": "4.2.0",
  "chai-as-promised": "7.1.1",
  "chai-bignumber": "3.0.0",
  "react": "16.8.4",
  "react-bootstrap": "1.0.0-beta.5",
  "react-dom": "16.8.4",
  "react-scripts": "2.1.3",
  "truffle": "5.0.5",
  "web3": "1.0.0-beta.55"
},
"scripts": {
  "start": "react-scripts start",
  "build": "react-scripts build",
  "test": "react-scripts test",
  "eject": "react-scripts eject"
},
```

```
  "eslintConfig": {
    "extends": "react-app"
  },
  "browserslist": [
    ">0.2%",
    "not dead",
    "not ie <= 11",
    "not op_mini all"
  ]
}
    {
      "constant": true,
      "inputs": [],
      "name": "name",
      "outputs": [
        {
          "name": "",
          "type": "string"
        }
      ],
      "payable": false,
      "stateMutability": "view",
      "type": "function"
    },
```

{
  "inputs": [],
  "payable": false,
  "stateMutability": "nonpayable",
  "type": "constructor"
}
"sourcePath": "C:/Users/ src/contracts/Smartcontracts.sol",
"ast": {
  "absolutePath": "/C/Users/
/src/contracts/Smartcontracts.sol",
  "exportedSymbols": {
    "Marketplace": [
      12
    ]
  },
  "id": 13,
  "nodeType": "SourceUnit",
  "nodes": [
    {
      "id": 1,
      "literals": [
        "solidity",
        "^",
        "0.5",

".0"
],
"nodeType": "PragmaDirective",
"src": "0:23:0"
},
{
"baseContracts": [],
"contractDependencies": [],
"contractKind": "contract",
"documentation": null,
"fullyImplemented": true,
"id": 12,
"linearizedBaseContracts": [
12
],
"name": "Marketplace",
"nodeType": "ContractDefinition",
"nodes": [
{
"constant": false,
"id": 3,
"name": "name",
"nodeType": "VariableDeclaration",
"scope": 12,

"src": "54:18:0",
"stateVariable": true,
"storageLocation": "default",
"typeDescriptions": {
  "typeIdentifier": "t_string_storage",
  "typeString": "string"
},
"typeName": {
  "id": 2,
  "name": "string",
  "nodeType": "ElementaryTypeName",
  "src": "54:6:0",
  "typeDescriptions": {
    "typeIdentifier": "t_string_storage_ptr",
    "typeString": "string"
  }
},
"value": null,
"visibility": "public"
},
{
"body": {
  "id": 10,
  "nodeType": "Block",

"src": "102:55:0",

"statements": [

{

"expression": {

"argumentTypes": null,

"id": 8,

"isConstant": false,

"isLValue": false,

"isPure": false,

"lValueRequested": false,

"leftHandSide": {

"argumentTypes": null,

"id": 6,

"name": "name",

"nodeType": "Identifier",

"overloadedDeclarations": [],

"referencedDeclaration": 3,

"src": "113:4:0",

"typeDescriptions": {

"typeIdentifier": "t_string_storage",

"typeString": "string storage ref"

}

},

"nodeType": "Assignment",

"operator": "=",
"rightHandSide": {
"argumentTypes": null,
"hexValue":
"4461707020556e69766572736974792d61726b6574706c616365",
"id": 7,
"isConstant": false,
"isLValue": false,
"isPure": true,
"kind": "string",
"lValueRequested": false,
"nodeType": "Literal",
"src": "120:29:0",
"subdenomination": null,
"typeDescriptions": {
"typeIdentifier":
"t_stringliteral_da4b6052d4e2657e92b9e8ac7ae34efe0016cdd765c6837ed00d7314b2dd8
},
"src": "113:36:0",
"typeDescriptions": {
"typeIdentifier": "t_string_storage",
"typeString": "string storage ref"
}
},
"id": 9,

"nodeType": "ExpressionStatement",
"src": "113:36:0"
}
]
},
"documentation": null,
"id": 11,
"implemented": true,
"kind": "constructor",
"modifiers": [],
"name": "",
"nodeType": "FunctionDefinition",
"parameters": {
"id": 4,
"nodeType": "ParameterList",
"parameters": [],
"src": "93:2:0"
},
"returnParameters": {
"id": 5,
"nodeType": "ParameterList",
"parameters": [],
"src": "102:0:0"
},

"scope": 12,
"src": "81:76:0",
"stateMutability": "nonpayable",
"superFunction": null,
"visibility": "public"
}
],
"scope": 13,
"src": "27:133:0"
}
],
"src": "0:160:0"
},
"legacyAST": {
"absolutePath": "/C/Users/
"exportedSymbols": {
"Marketplace": [
12
]
},
"id": 13,
"nodeType": "SourceUnit",
"nodes": [
{

```
  "id": 1,
  "literals": [
    "solidity",
    "^",
    "0.5",
    ".0"
  ],
  "nodeType": "PragmaDirective",
  "src": "0:23:0"
},
{
  "baseContracts": [],
  "contractDependencies": [],
  "contractKind": "contract",
  "documentation": null,
  "fullyImplemented": true,
  "id": 12,
  "linearizedBaseContracts": [
    12
  ],
  "name": "Marketplace",
  "nodeType": "ContractDefinition",
  "nodes": [
    {
```

"constant": false,

"id": 3,

"name": "name",

"nodeType": "VariableDeclaration",

"scope": 12,

"src": "54:18:0",

"stateVariable": true,

"storageLocation": "default",

"typeDescriptions": {

"typeIdentifier": "t_string_storage",

"typeString": "string"

},

"typeName": {

"id": 2,

"name": "string",

"nodeType": "ElementaryTypeName",

"src": "54:6:0",

"typeDescriptions": {

"typeIdentifier": "t_string_storage_ptr",

"typeString": "string"

}

},

"value": null,

"visibility": "public"

```
},
{
  "body": {
    "id": 10,
    "nodeType": "Block",
    "src": "102:55:0",
    "statements": [
      {
        "expression": {
          "argumentTypes": null,
          "id": 8,
          "isConstant": false,
          "isLValue": false,
          "isPure": false,
          "lValueRequested": false,
          "leftHandSide": {
            "argumentTypes": null,
            "id": 6,
            "name": "name",
            "nodeType": "Identifier",
            "overloadedDeclarations": [],
            "referencedDeclaration": 3,
            "src": "113:4:0",
            "typeDescriptions": {
```

"typeIdentifier": "t_string_storage",
"typeString": "string storage ref"
}
},
"nodeType": "Assignment",
"operator": "=",
"rightHandSide": {
"argumentTypes": null,
"hexValue":
"4461707020556e6976657273697479204d61726b6574706c616365",
"id": 7,
"isConstant": false,
"isLValue": false,
"isPure": true,
"kind": "string",
"lValueRequested": false,
"nodeType": "Literal",
"src": "120:29:0",
"subdenomination": null,
"typeDescriptions": {
"typeIdentifier":
"t_stringliteral_da4b6052d4e2657e92b9e8ac7ae34efe0016cdd765c6837ed00d7314b2dd8f36",
"typeString": "literal_string \"
},

```
          "value": "
        },
        "src": "113:36:0",
        "typeDescriptions": {
          "typeIdentifier": "t_string_storage",
          "typeString": "string storage ref"
        }
      },
      "id": 9,
      "nodeType": "ExpressionStatement",
      "src": "113:36:0"
    }
  ]
},
"documentation": null,
"id": 11,
"implemented": true,
"kind": "constructor",
"modifiers": [],
"name": "",
"nodeType": "FunctionDefinition",
"parameters": {
  "id": 4,
  "nodeType": "ParameterList",
```

```
        "parameters": [],
        "src": "93:2:0"
      },
      "returnParameters": {
        "id": 5,
        "nodeType": "ParameterList",
        "parameters": [],
        "src": "102:0:0"
      },
      "scope": 12,
      "src": "81:76:0",
      "stateMutability": "nonpayable",
      "superFunction": null,
      "visibility": "public"
    }
  ],
  "scope": 13,
  "src": "27:133:0"
 }
],
```

APPENDIX 4: Program Code for Arduino Microcontroller

```
        ORG 00H
        LJMP MAIN
        ORG 300H
TBL:    DB 0C0H,0F9H,0A4H,0B0H,99H,92H,82H,0F8H,80H,90H    ;7seg data for
comm. anode type
        ORG 30H

MAIN:   MOV P2,#00H
        MOV P3,#00H
        ACALL FRONT
        MOV DPTR,#TBL
        CLR A
        MOV 40H,#10
        MOV 43H,#10
        MOV 46H,#20
        MOV 49H,#20
        MOV R0,#35
        MOV R6,#30
        MOV R7,#40

X1:     MOV A,40H
        MOV B,#10
        DIV AB
        MOV 41H,A
        MOV 42H,B


A1:     SETB P3.0
        CLR P3.1
        MOV A,41H
        MOVC A,@A+DPTR
        MOV P2,A
        ACALL DELAY
        MOV P3,#00H
        SETB P3.1
        CLR P3.0
        MOV A,42H
        MOVC A,@A+DPTR
        MOV P2,A
       ACALL DELAY
        MOV P3,#00H
        SJMP X3
X2:     SJMP X1
X3:     MOV A,43H
        MOV B,#10
        DIV AB
        MOV 44H,A
        MOV 45H,B
        SETB P3.2
        CLR P3.3
        MOV A,44H
```

```
MOVC A,@A+DPTR
MOV P2,A
ACALL DELAY
MOV P3,#00H
SETB P3.3
CLR P3.2
MOV A,45H
MOVC A,@A+DPTR
MOV P2,A
ACALL DELAY
MOV P3,#00H


MOV A,46H
MOV B,#10
DIV AB
MOV 47H,A
MOV 48H,B
SETB P3.4
CLR P3.5
MOV A,47H
MOVC A,@A+DPTR
MOV P2,A
ACALL DELAY
MOV P3,#00H
SETB P3.5
CLR P3.4
MOV A,48H
MOVC A,@A+DPTR
MOV P2,A
ACALL DELAY
MOV P3,#00H


MOV A,49H
MOV B,#10
DIV AB
MOV 50H,A
MOV 51H,B
SETB P3.6
CLR P3.7
MOV A,50H
MOVC A,@A+DPTR
MOV P2,A
ACALL DELAY
MOV P3,#00H
SETB P3.7
CLR P3.6
MOV A,51H
MOVC A,@A+DPTR
MOV P2,A
ACALL DELAY
MOV P3,#00H
```

```
         DJNZ R0,X2
         MOV R0,#35

         DJNZ 40H,Q1
         MOV 40H,#20

Q1:      DJNZ 43H,Q2
         MOV 43H,#10
         ACALL RIGHT

Q2:      DJNZ 46H,Q3
         MOV 43H,#20
         MOV 46H,#10

Q3:      DJNZ 49H,Q4
         MOV 49H,#10
         ACALL BACK

Q4:      DJNZ R6,X4
         ACALL LEFT
         MOV 40H,#10
         MOV 43H,#10
         MOV 46H,#30

X4:      DJNZ R7,L1
         LJMP MAIN
L1:      LJMP X1

DELAY:   MOV R4,#5
H2:      MOV R5,#0FFH
H1:      DJNZ R5,H1
         DJNZ R4,H2
         RET

FRONT:   MOV P1,#54H
         MOV P0,#02H
         RET

RIGHT:   MOV P1,#0A1H
         MOV P0,#02H
         RET

BACK:  MOV P1,#09H
       MOV P0,#05H
       RET

LEFT:  MOV P1,#4AH
       MOV P0,#08H
       RET
```