TRAFFIC CONTROL STRATEGY FOR ADAPTIVE SIGNAL CONTROLLER BASED ON REINFORCEMENT LEARNING AND LOCAL COMMUNICATION CHANNEL

MUAID ABDULKAREEM ALNAZIR AHMED

DOCTOR OF PHILOSOPHY (ENGINEERING)

LEE KONG CHIAN
FACULTY OF ENGINEERING AND SCIENCE
UNIVERSITI TUNKU ABDUL RAHMAN
AUGUST 2023

TRAFFIC CONTROL STRATEGY FOR ADAPTIVE SIGNAL CONTROLLER BASED ON REINFORCEMENT LEARNING AND LOCAL COMMUNICATION CHANNEL

By

MUAID ABDULKAREEM ALNAZIR AHMED

A thesis submitted to the Department of Civil Engineering,
Lee Kong Chian Faculty of Engineering & Science,
Universiti Tunku Abdul Rahman,
in fulfilment of the requirements for the degree of
Doctor of Philosophy (Engineering)
August 2023

DEDICATION

The work is dedicated to scholars and researchers who seek knowledge and cognition.

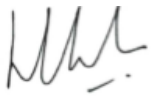
ABSTRACT

This research study is in the field of deep reinforcement learning (DRL) adaptive controllers. The developed DRL controller is an off-policy, model-free agent based on the Q-learning algorithm. The research aims to address several issues found in the existing DRL work direction. Issues related to the ability of the DRL agent to mitigate signal operation under various traffic flow conditions, the extension of the model environment in the development process of the DRL agent, the under-representation and simplification of traffic dynamics, the utilisation of futuristic communication technology, and the ability of the DRL system to mitigate signalised junctions in an arterial network are pressing challenges for intelligent signal systems. An innovative control strategy is proposed to make the single system design efficient for global optimisation at network-level operation. The introduced downstream policy adapts the signal operation to the available capacity at discharge routes. An illustrative case study tests and evaluates the proposed control system. The micro-model simulated stochastic and dynamic traffic elements to represent the actual traffic. The rigorous tests showed that the proposed controller achieved the closest optimal flow condition at 0.80 for the network operation and outperformed other controllers in reducing waiting time costs (10%-36%), improving travel time experiences (5%−25%), and constituting the highest mean travel speed (3.4 m/s).

APPROVAL SHEET

This thesis entitled "TRAFFIC CONTROL STRATEGY FOR ADAPTIVE SIGNAL CONTROLLER BASED ON REINFORCEMENT LEARNING AND LOCAL COMMUNICATION CHANNEL**"** was prepared by MUAID ABDULJAREEM ALNAZIR AHMED and submitted as fulfilment of the requirements for the degree of Doctor of Philosophy (Engineering) at Universiti Tunku Abdul Rahman.

Approved by:



_____
(Prof. Ir. Dr. KHOO HOOI LING)
Date:…21/08/2023……..
Professor/Supervisor
Department of Civil Engineering
Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

_____
(Asst. Prof. Dr. NG OON-EE)
Date:… 21/08/2023……..
Assistant Professor/Co-supervisor
Department of Mechatronics and BioMedical
Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

DECLARATION

I hereby declare that the dissertation is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UTAR or other institutions.

Name Muaid Abdulkareem Alnazir Ahmed

Date _20/08/2023_____

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

S	Traffic flow
S_{cl}	Controlled traffic flow
$S_{cl.N}$	Controlled inbound traffic flow from northbound of junction
S_{ucl}	Uncontrolled traffic flow
$S_{ucl.N}$	Uncontrolled inbound traffic flow from northbound of junction
S_{ext}	Exiting traffic flow
$S_{ext.Z2}$	Exiting traffic flow at Z2 of junction
$S_{ext.N}$	Exiting traffic flow heading northbound of junction
$S_{opt.dwn}$	Downstream optimum traffic flow
S_{dis}	Discharge traffic flow at Z3 of junction
$S_{dis.N}$	Discharge traffic flow heading northbound of junction
L	Distance from traffic signal stop line
l	Detection area length
p_{exe}	Executive signal phase
v	Speed
v_{ent}	Entering speed of vehicles to Zone 1 of intersection
v_{ext}	Exiting speed of vehicles from Zone 1/ Entering speed of vehicles to Zone 2
v_{merg}	Merging speed at Zone 2 due to impact from uncontrolled traffic flow/Exiting speed of vehicles from Zone 2/Entering speed of vehicles to Zone 3
v_{depart}	Departing speed of vehicles from Zone 3/ Exiting speed of vehicles

	from Zone 3
v_{lmt}	Speed limit
v_f	Free flow speed
V_{tot}	Total traffic volume
V	Traffic volume
V_{cl}	Traffic volume at controlled lanes
V_{ucl}	Traffic volume at uncontrolled lanes
V_{max}	Maximum traffic volume
k	Density
$k_{w.dwn}$	Weighted density at downstream discharge zone
k_{upstrm}	Density at upstream approach lanes within l distance from stop line
k_{Z3}	Density at Z3 section of signalled junction
k_{opt}	Optimum density
k_{jam}	Jam density (2 k_{opt})
D_{green}	Green time phase duration
D_{eff}	Effective green time duration (T_{green} + T_{yellow})
C_{sec}.	Capacity of road section
$C_{sec.Z3}$	Capacity of road section at Z3
C_{opt}	Optimum cycle length
Δv_{Z2}	Speed difference at Zone 2 of signalled junction (v_{merg}- v_{ext})
Δv_{Z3}	Speed difference at Zone 3 of signalled junction (v_{depart}- v_{merg})
T_{lst}	Start-up lost time
T_{min}	Minimum time duration
h	Headway time gap
T_{green}	Green time duration

T_{yellow}	Yellow time duration
T_{max}	Maximum time duration
R	Scalar quantity for Total reward gains ($R_k + R_s$)
R_k	Scalar quantity to measure optimisation for density at intersection
R_s	Scalar quantity to measure optimisation for speed at intersection ($\|\Delta v_{Z2}\|$ +CV)
$\|\Delta v_{Z2}\|$	Scalar quantity of speed difference for $S_{ext.Z2}$
CV	Coefficient of variation (σ/μ)

CHAPTER 1

INTRODUCTION

1.1 Background

Traffic light signals are signalised devices positioned at road intersections to manage traffic flows. Traffic signal operation is a unique endeavour that falls within traffic management. The exponential growth in urban traffic count has caused traffic management to become more complex. As the investment option to expand the road infrastructure is not socially feasible (Balaji et al., 2010) and is restricted to financial capacity and space availability (Vidali et al., 2019), congestion continues to grow. This urban traffic congestion adversely impacts society, the economy, and the environment (Ali et al., 2021). Therefore, it is of paramount importance to optimise the existing road network using signal control systems.

Traffic control systems have experienced tremendous development in control strategies in the past 100 years. Three (3) major traffic light control generations include fixed time, actuated and adaptive control systems. Fixed time or pre-timed control systems are designed based on historical data to create rigid timing plans stored in a control unit's time clock. Since the development of the pre-timed controller in the early 1900s, it has widely spread and is still commonly used today. The fixed controller is simple in

design and cost-efficient; nonetheless, the system fails to accommodate the stochastic nature of traffic and responds to uncertainty in traffic conditions. The system assumes constant traffic flows and cannot deal with disruption and unaccounted fluctuation.

The limitations of fixed systems paved the way for the second generation of controllers, namely actuated traffic signals, in the late 1920s (Gordon and Tighe, 2005). The actuated controllers rely on real-time data from on-site installed sensors such as pressure detectors, loop detectors, radar, and video visuals to make control decisions. The control decisions are related to phase calling, green extension, gap out, and max out decisions (Feng et al. 2015). The responsive system works well for isolated intersections; nonetheless, the actuated logic performance fails at low-speed saturated intersections such as congested grid intersections. Besides that, any control decision made by the actuated controller will occur in the following signal phase, causing a delayed response for past events.

In order to close the gap between the present traffic needs and signal operation, adaptive controllers emerged. The advantage of the adaptive controller is that it can forecast the traffic condition and act accordingly to address the need in the future. The developments of microprocessors in the 1960s revolutionised the aspect of controllers by making them more decision-makers (Gordon and Tighe, 2005). The adaptive controller does not require an exact cycle length, in contrast with the actuated control system (Gordon and

Tighe, 2005). The adaptive controller requires minimal expert intervention to adjust the phase timing compared to the fixed controller.

Meanwhile, adaptive strategy adheres to challenging theories such as traffic flow and stochastic traffic environment, as the system does not require a complete understanding of its surrounding. Overall, the third generation of signal controllers has the potential to achieve higher time-saving on roads, lower human intervention, and faster adaptation to the stochastic nature of traffic behaviour. Table 1.1 presents a summary of the traffic signal controllers.

Table 1.1: Traffic signal control generations (modified from Gordon and Tighe, 2005)

Type of Controller	Introduction	Traffic Environment			Limitation
		Isolated	Arterial	Grid	
Pre-timed	1900s	Not appropriate	Appropriate [requirement: coordination & traffic flow hierarchy]	Appropriate	Assumption of constant traffic flow
Actuated	1920s	Appropriate	Appropriate [requirement: coordination, high-speed traffic flow, detectors setting >40m]	Not appropriate	Delayed response
Adaptive	1960s	Effective[#]			Representation of environment Complex design

[#]Experimental tests indicate that the system can work in a grid environment

In the past two decades, adaptive controllers have been widely researched and examined. Many theories are proposed and studied, but the implementation and deployment of third-generation controllers are still limited

and restricted. There are a few issues facing the deployment of adaptive systems, including control strategy and infrastructure readiness. By far, no adaptive research has focused on proposing an innovative control strategy. The present research direction implements a responsive strategy to address traffic operations. Such a focus area eventually forces the system developers to rely on an accurate representation of the environment. Their responsive systems fail to address the intersection capacity and utilisation of available road spacing to control the traffic flow.

There are various algorithms used to construct adaptive controllers. However, among all the adaptive signal controllers using ML and AI techniques, reinforcement learning (RL) is nominated as the favourable method for architecting the adaptive signal controller (Ma et al., 2016). This class of ML has the capacity for self-learning to extrapolate correct actions in an interactive problem. The interactive problem is a form of environment where the appropriate behaviour for every condition is usually impractical to be constituted entirely (Sutton and Barto, 2018). This category of problem is found in traffic movement. Besides that, this characteristic differentiates RL from supervised learning. The latter method is limited to expert knowledge. Furthermore, unlike unsupervised learning, RL utilises learning (trial and error) based on a reward signal to treat the problem.

Many researchers have proposed systems that assume connected vehicle technology to access the road environment. There are two (2) drawbacks to this assumption. First, roads and vehicles are yet to be equipped

with such technology. Second, the investment in these technologies is proven expensive. It will require intervention and collaboration among different parties. Therefore, implementing connected vehicles is expensive, and the preference in practice is given to traffic light systems based on information from existing deployed infrastructure (Jin, 2018). On the other hand, other researchers have looked into developing adaptive systems based on the status quo of present infrastructure detection devices and vehicles on the road.

As traffic congestion is today's problem, the focus of this research is to contribute to developing a deep reinforcement learning (DRL) controller system based on available detection devices rather than a futuristic assumption in the form of connected vehicle communication. The utilisation of present detection technology is important to make the intelligent system practical and applicable for deployment. In addition, we introduce a new traffic control strategy for signal operation. The control technique enhances the capability of the DRL controller to adapt to traffic dynamics and utilise available capacity to optimise signal operation.

1.2 Problem Statement

Despite the fruitful developments of models and solution techniques in adaptive traffic signal control, some significant limitations in this thesis have been examined and tackled to add a genuine and practical contribution to the field of traffic engineering. These limitations are addressed in the following points.

1. **Environment Setting and Evaluation**: The simplicity of studied intersections in researching the adaptive controller. Simple geometrical layouts of junctions. Traffic flows are constant, low, and have a hierarchy. Uncontrolled turning movements are not enclosed. Short-time plan and fixed-phase design. Uniform behaviour of road users. As traffic environment stochasticity is under-represented accurately, the valuation of DRL controllers in mitigating traffic flows remains in question.
2. **Controller Design and Communication Protocol**: The current DRL system designs rely heavily on accurately representing the environment. The efficiency of the DRL controller drops whenever the communication channel delivers partial observation. In this context, studies have constantly reported that the DRL agent has either performed equivalently to conventional systems (i.e., fixed and actuated signal controllers) or even worse. Hence, developing an efficient agent using only boundary-based observations within the intersection level is yet to be achieved.
3. **Traffic Control Strategy:** Though DRL agents are intelligent systems, they rely on reward signals to guide their decisions. The reward, in this case, is the traffic control strategy. Current researchers utilise responsive strategies related to vehicle features. The extracted features have a little advantage as they serve particular instances under certain traffic conditions (e.g.,

low demand and hierarchal directional flow). Studies have reported that the efficiency of DRL control drops as the traffic environment experiences unstable conditions. Hence, there is an urgent need to develop traffic control techniques to advance the DRL application.

Collectively, the current limitations question the **scalability** of the DRL signal logic for real-world applications.

1.3 Research Questions

Based on the current development in the field of adaptive controllers, the following questions are relevant to the context of the limitations addressed in this thesis.

1. What is the extension of the environment model on the agent's performance?
2. Can the deep learning agent compete against other signal system techniques if detection zones are restricted?
3. How to develop a single agent system that is efficient for network operation?
4. What is a comprehensive traffic control strategy for different environmental settings in relation to mixed modes of vehicular flow, geometric configuration, and intersection design?

1.4 Research Objectives

This research aims to develop an adaptive signal controller system using a deep reinforcement learning (DRL) approach. The controller's design must consider the practical aspects of development and current infrastructure readiness. This core goal is divided into a number of objectives, including:

1. To develop a deep Q-learning controller agent for stochastic traffic conditions.
2. To design an adaptive controller agent based on existing communication technology and a defined detection zone.
3. To test the application of a single agent system and its efficiency in a traffic network operation context.
4. To measure the effectiveness of traffic control policies for network-wide adaptive signal systems.

1.5 Scope of the Research

The study aims to develop an adaptive traffic control system capable of operating at different levels of road intersections. At the most superficial level of the network is to optimise a single isolated intersection. A more complex road structure of connected intersections to form an arterial road network is researched at the second level. The information inputs are based on real-time traffic data from the studied road junctions. The experimental setting is based on actual network configuration, where calibration and validation procedures are emphasized to accurately replicate driving and traffic conditions. The

replicated conditions are road geometry, junction configuration, time plan, traffic composition, and driver's behaviour. Stating that all the experimental analyses are carried out in a simulation environment. Testing the developed software in a non-model environment is not part of this research due to the associated implications and restrictions of deploying experimental systems in actual life conditions.

As the study is carried out in a model environment, it is assumed that built-in infrastructure detectors are available on-site. Though this assumption might conflict with the real traffic environment, it is also believed that acquiring and installing detection devices is attainable in real-world applications. In this research study, detection devices are lanearea and loop detectors. These hardware devices are commonly and widely used in signal operation.

1.6 Outline of the Thesis

Besides the introduction chapter herein, the remainder of the thesis is divided into seven (7) chapters, including (i) Literature Review, (ii) System Features and Control Policy, (iii) Methodology, (iv) Results and Discussion, (v) Conclusion, (vi) List of References, and (vii) Appendices.

In the Literature Review (Chapter 2), an extensive review was carried out for the adaptive control studies and the more recent developments in the area of interest, i.e., deep reinforcement learning. Chapter 3 (System Features

and Control Policy) presents the technique and system features for the proposed signal logic. Chapter 4 (Methodology) details the procedure in terms of data collection, micro-model development, control strategy design, deep reinforcement logic and training, and comparative control systems and evaluation environments. Chapter 5 (Results and Discussion) reports the associated outputs of the proposed control logic and compares the findings against other comparative systems. Chapter 6 (Conclusion) summarises the thesis work and recommends future work direction. Chapter 7 (List of References) lists the research works that are cited in this thesis. Chapters 8 to 11 (Appendices) are needed to provide a separate discussion and details in relation to the methodology chapter. Appendix E contains the student's biography and list of publications.

CHAPTER 2

LITERATURE REVIEW

The literature review contains studies on the aspect of adaptive controller generation. Nearly 60 studies were carefully selected for this review. The intelligent controller systems are categorised based on (i) design (Section 2.2), (ii) communication protocol (Section 2.3), (iii) optimisation technique (Section 2.4), and (iv) algorithmic approach (Section 2.5). At the end of each section, a discussion is presented to point out specific challenges within a class. Section 2.6 assesses common challenges and limitations of current adaptive controller studies. Unlike the specific challenges, the joint review intends to connect topics across the different classes of adaptive programmes. In Section 2.7, prominent deep reinforcement learning (DRL) studies are categorised and detailed based on five (5) DRL architectures. The current limitations of DRL research are extended in Section 2.8.

2.1 Adaptive Traffic Light Signal Controller

The adaptive controller can close the gap between near-future needs and signal operation. This feature distances the adaptive controller from the earliest forms of controllers (fixed and actuated systems). Adaptive controllers have been widely researched and examined in the past two decades. The following sub-sections categorise adaptive signal systems based on system

design for signal operation, physical communication channels, traffic control strategies, and algorithm techniques. Figure 2.1 presents the categorization of adaptive controllers in this literature review study.

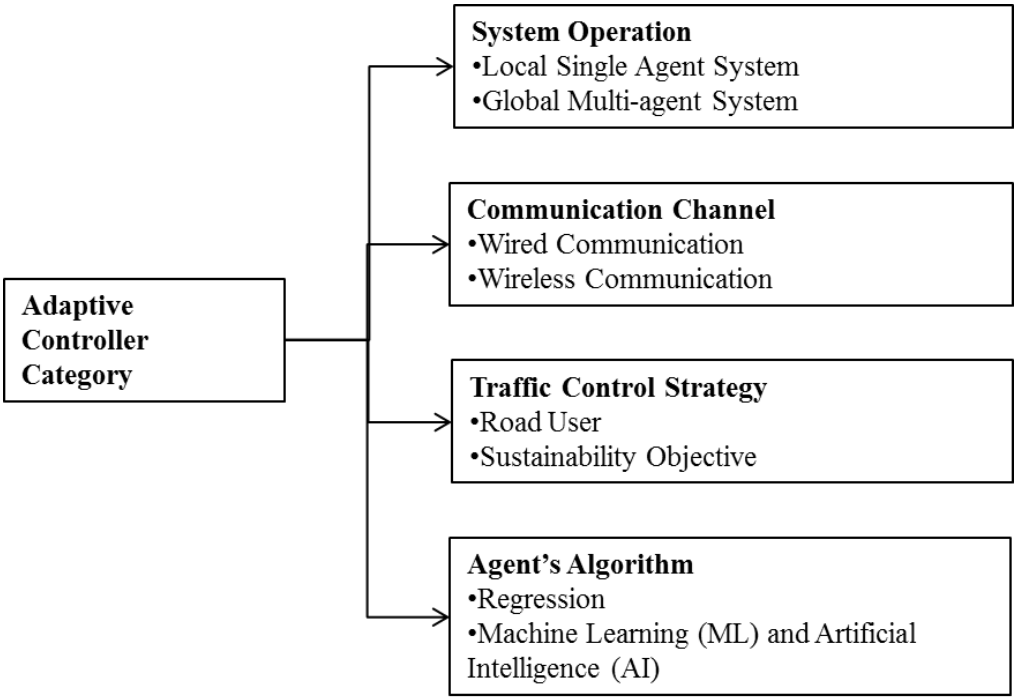


Figure 2.1: Categories of adaptive traffic light control

2.2 Classification Based on Traffic Light Signal Operations

Traffic signal operation is a unique endeavour that falls within traffic management. The impacts of signal operations are usually underestimated, despite the improved technology. An urban road network's capacity can be increased by using traffic signal controllers (Abdoos et al., 2013; McKenney and White, 2013). Adaptive control systems consider measured and predicted traffic data input variables to utilise infrastructure capacity. The gathered data are treated and processed to achieve local and global optimisations.

Local optimisation is a decentralised controller technique where the agent acts independently (in solo) to acquire the best performance at the intersection level. These systems can adapt to the demand at the intersection level and typically have simple logic. These logics do not require interference with neighbouring intersections (Płaczek, 2014). In addition, these single-system designs have primarily focused on flow theory and proposed solutions regarding scheduling departures at intersections and capacity allocation.

In contrast, global optimisation is a centralised controller technique where the executed action is placed after considering the network input. The multi-agent system assigns a single agent to each intersection, allowing agents to 'tutor' each other to decide on optimal plans for small areas (McKenney and White, 2013). The centralised approach makes NP-hard a problem, which can lead to high computational complexity for real-time deployment (Płaczek, 2014). The control policies of multi-agent systems focus on vehicle progression, limiting the system's efficiency by relying on complete observation of the traffic environment.

2.2.1 Local Control System Design

Varaiya (2019) proposed an adaptive maximum pressure (AMP) to control a traffic signal. The AMP policy is based on the product of weighted queue length and corresponding saturation flow for each phase. The control algorithm intends to maximise the throughput at the intersection level. The major limitation of the proposed policy is that it assumes infinite storage

capacity in each lane and does not consider a ‘de facto red’ movement where finite queues are observed or when shared movements are within the same lane. The AMP is a greedy algorithm that leads to a locally optimal solution (Wei et al., 2019).

Hao et al. (2018) designed an online local model predictive controller (LMPC) using Urban Cell Transmission Model (UCTM) system feedback for traffic density. The LMPC agent forecasts the local delay and decides on the switching time to minimise the delay for all vehicles over a prediction horizon of a few minutes. The LMPC system was compared against the max-pressure and pre-timed controllers. The findings showed that LMPC performed best in heavy traffic conditions and recorded the least cumulative delay. In terms of outflow, all systems had similar output. On the other hand, the proposed controller's performance deteriorated with inaccurate traffic condition prediction.

Tiaprasert et al. (2015) formulated a queue-based adaptive control system. The proposed system relies on connected vehicle communication to estimate the queue length. The proposed model does not require signal timing, traffic volume, or queue characteristics as inputs. A discrete wavelet transform (DWT) was utilised to improve the steadiness of queue estimation regardless of the penetration ratio. The proposed method was implemented without assuming a time plan or a specific arrival distribution. The micro-simulation results indicated that the DWT is able of estimating queue length for various

flow conditions (under-saturated and saturated). Therefore, DWT could enhance the performance of an adaptive traffic control system.

Gregoire et al. (2014) studied back-pressure control based on queue capacity (BPC). The system's goal is to normalise pressure across all queues and allow high upstream pressure to flow to low downstream pressure. The normalisation procedure is expected to decrease the blocking probability. The authors utilised downstream capacity to regulate traffic flow from upstream. For testing, the authors simulated an isolated intersection with four (4) flow scenarios. The results showed that the BPC system had the least average travel time in the high traffic flow condition compared to the back-pressure (Wongpiromsarn, 2012) and fixed systems. As in other back-pressure systems, BPC suffers from the "last packet problem" (Ji et al., 2012). A vehicle may be starved for a long time if other routes have more considerable queue lengths.

Goh et al. (2012) proposed an online map-matching algorithm based on the Hidden Markov Model (HMM). The study focused on two (2) improvements over existing HMM-based algorithms, including (1) the use of the variable sliding window (VSW) method to guarantee quality solutions under uncertain future inputs and (2) the novel combination of spatial, temporal, and topological information using machine learning. The study evaluated the accuracy of the algorithm using field test data. The results indicated that the VSW outperformed the traditional localising method in terms of both accuracy and output delay. Moreover, the results suggest that VSW is viable for low-latency applications such as traffic sensing.

2.2.2 Global Control System Design

Wang et al. (2020) proposed an arterial coordinated real-time adaptive control model. The method aims to move vehicle platoons along an arterial road with the least signal delay and higher throughput. The system develops a joint control to optimise the speed of connected vehicles and coordinate signals along the arterial simultaneously. The cooperative controller manages a phase length duration to run the vehicle platoon. The authors tested the system in a traffic model comprising five (5) signalled intersections. The MAXBAND algorithm was tested against the proposed model. The results showed that the proposed joint controller reduced the number of stops by 53.69% and the stops of coordinated signals by 41.15%. The study assumed a 100% fully connected vehicle environment.

Darmoul et al. (2017) proposed an immune network algorithm based multi-agent (INAMAS) for an adaptive control system. The heterarchical architecture assigned one (1) agent to each signalled intersection. Each agent communicates and coordinates with neighbouring agents to share experiences and adapt to road disturbances. The authors limit the communication to only two (2) agents (neighbours). To demonstrate the efficiency of the INAMAS, the authors tested the system in various scenarios of traffic conditions for three (3) and six (6) signalled intersections models. The INAMAS outperformed comparative systems: fixed and longest-queue-first-maximal weight-matching (LQF-MWM) algorithms in terms of average total delay and average queue length.

Liu et al. (2017) presented a dynamic clustering algorithm for cooperative reinforcement agent (CRLFA). The clustering algorithm is based on enhanced affinity propagation to ensure stability. A fast gradient-descent function approximation is utilised to seek optimal policy and mitigate the curse of dimensionality associated with reinforcement learning. The system is tested in a model network environment and compared against three (3) controllers: fixed timing, longest-queue-first algorithm (LQFA) and classical reinforcement learning (RL). The results indicated that the proposed method surpassed the traditional adaptive signal control method in improving throughput, reducing waiting time, and avoiding traffic congestion.

Kari et al. (2014) presented an online agent-based adaptive traffic signal control based on queue length (ATSC). The proposed methodology considers two (2) agents at an intersection, including (1) vehicle agent (VA); responsible for communicating real-time vehicle data, and (2) intersection management agent (IMA); undertakes to communicate with all VA within a communication radius to determine the optimal signal timing. The authors tested the system in a single isolated intersection with two (3) traffic flow scenarios (constant and varied). The results indicated that the system achieved moderate savings of 5% to 14% in reducing travel time and 0% to 5% savings in fuel consumption for the constant demand scenario. The system scored significant travel time and fuel consumption savings in varied demand scenarios at 61% and 32%, respectively.

Xiang and Chen (2016) presented a novel multi-agent control method for an integrated network of adaptive traffic signal controllers in a vehicle-to-intersection (V2I) communication environment. The system has two (2) innovations: (i) the utilisation of the Gauss model for parallel processing; and (ii) the provision of co-learning to recommend the shortest time path. The intersection is treated as an agent. Further, the agents interacted with the environment by trying out actions and using the resulting feedback to reinforce behaviour for the desired outcome. The simulation results showed that measured vehicular attributes (travel time, delay, and queue length) under the proposed signal system were reduced significantly compared to the traditional traffic signal. The limitation is that the system assumes a Markov decision process to model an agent's intersection with its own environment.

2.2.3 Review of System Design

The increasing demand for mobility in the 21st century poses a challenge in dealing with traffic and transportation systems. Therefore, efficient management tools and techniques are required to deal with urban traffic in terms of control, optimised use of existing infrastructure, efficient assignment of the demand, and so on (Bazzan and Klügl, 2014). The system functionality is designed to achieve a local optimisation at the intersection level or a global optimisation at the network level. Global optimisation is approached using multi-agent systems. The review highlighted two (2) standard techniques for multi-agent controllers: centralised control and coordinated control.

The centralised control is a hierarchical multi-agent. In the first level, agents representing intersections (low level) are responsible for sending traffic state to a higher level. The highest level is the coordinator agent, which provides optimal local signal plans after evaluating all network traffic states from low-level agents. Such centralised design ultimately leads to crippling the operation due to the exponential amount of information and variables that require the attention of the centre agent. This limitation is not addressed clearly, and how to practically implement this system mechanism in mitigating a large-scale real traffic network is not answered yet.

In order to provide flexibility and robustness, some studies favoured the coordination approach for global optimisation. The coordination framework allows agents to exploit known configurations (local call) and simultaneously explore action combinations (with neighbouring intersections) to attempt better gains. The coordination is often restricted to adjacent intersections only to reduce messages and computational resources. The coordinated controllers do not seem to produce a concrete solution for conflicting optimisation. This is a case where the neighbours' request is incompatible with the local agent’s action to solve its local optimisation problem (Bazzan and Klügl, 2014).

Studies carrying out a comparison between local and global control techniques are rare. Among these rare findings is Brys et al. (2014), who presented two controllers: coordinated DCEE and RL-SARSA controllers. The authors stated that coordination among agents is not necessarily beneficial;

SARSA (single control) surpassed DCEE (multi-agent control). Unlike global optimisation, the local approach is more straightforward and can be deployed in an extensive scale network. The following Table 2.1 presents a summary based on system design methods.

Table 2.1: Classification of adaptive control studies based on system design

System Design	Adaptive Traffic Control Study
Local Control (Single Agent)	Varaiya (2019), Wang et al. (2019), Yao et al. (2020), Hnaif et al (2019), Deligkas et al. (2018), Hao et al. (2018), Gao et al. (2017), Zaidi et al. (2016), Tiaprasert et al. (2015), Płaczek (2014), Gregoire et al. (2014), Wongpiromsarn et al. (2012), Putha et al. (2012)
Global Control (Multi-agent)	Galvan-Correa et al. (2020), Rasheed et al. (2020), Tan et al. (2019), Li et al. (2018), Zhou et al. (2017), Darmoul et al. (2017), Ma et al. (2016), Mannion et al. (2015), Kari et al. (2014), Khamis and Gomaa (2014), Smith et al. (2013), He at al. (2012)

2.3 Classification Based on Physical Communication Channel

An intersection contains hardware devices to gather local traffic statistics. The adaptive control system's quality depends on the available on-road vehicle data. The ratio of the input data known to the adaptive control system is known as the penetration rate (Priemer and Friedrich, 2009). Vehicle data such as speed, location, flow, density, and number of vehicles within road infrastructure are vital for the control strategy. Two (2) sources for vehicular information include infrastructure built-in detection devices and wireless communication channels.

2.3.1 Wired Communication Channel based Adaptive Controller

Some of the designed adaptive controllers rely on utilising existing infrastructure sensors. Sensors such as in-pavement, video-based loop, and

micro-wave detectors are standard methods to collect data for system controllers. The advantage of this assumption in developing a control system is that little investment is required to improve road infrastructure. On the other hand, these data input devices can only provide instantaneous vehicle information such as location, speed, and acceleration (Feng et al., 2015). This limitation deteriorates the effectiveness of the adaptive system, which requires extensive data input in the absence of a forecasting technique for vehicle attributes.

Hnaif et al, (2019) proposed an intelligent road traffic management system based on a human community genetic algorithm (IRTMS). The IRTMS system is built on the number of vehicles obtained at the intersection level using an infrared sensor. The proposed intelligent controller aims to optimise vehicle variables, including speed, density, and traffic volume, via communicating with neighbouring junctions to obtain a data feed. The authors reported that the IRTMS system had the minimum waiting time and total time compared to the current conventional system.

Deligkas et al. (2018) designed a schedule-driven adaptive traffic controller based on the value of time (VoT). The heuristic forward search algorithm intends to minimise the total time cost for cars waiting at an intersection. The authors tested the system in two (2) configurations of cross-intersection (simple and complex). The authors compared the proposed controller against a dynamic programming approach. The results showed that

the VoT policy is a better optimisation technique than flow attributes in asymmetric traffic situations.

Płaczek (2014) introduced a self-organizing traffic light (SOTL) system for urban traffic network. The agent optimises signal control based on incoming traffic to an intersection. Then the SOTL algorithm predicts the effect of possible control actions on the delay of all vehicles. Based on the predicted total delay of all vehicles, a control decision (green phase) is made for a traffic stream(s) with a minimum delay (lowest action cost). Płaczek (2014) experimented against other SOTL systems in Gershenson (2005) and Helbing et al. (2005) studies. The results indicated that the proposed SOTL outperformed other systems. Besides that, the authors stated the reliance of the SOTL on the vehicle position information.

Smith et al. (2013) proposed a scalable urban traffic control (SURTRAC) system. The SURTRAC is a multi-agent planning computed schedule to decide green phase switching using flow rate. Then the forecasted traffic outflows are communicated to downstream neighbouring intersections to increase the visibility of vehicles and assist in their respective planning. A pilot implementation was carried out for the SURTRAC in nine (9) intersection network in Pittsburgh, Pennsylvania, US. The SURTRAC was compared against a site actuated system. The actuated timing plans were generated using the SYNCHO offline timing package. The findings indicated that the proposed system significantly reduced vehicles' travel time (26%) and emissions (21%).

2.3.2 Wireless Communication Channel based Adaptive Controller

The built-in vehicular technologies and intelligent infrastructure advancements have offered new vehicle detection opportunities in traffic network environments (Feng, 2015). The new wireless communication is known as the connected vehicle (V2X), where a vehicle communicates with other vehicles (V2V) and with the infrastructure (V2I) within dedicated short-range communication (DSRC) (Wang et al., 2018, Feng et al., 2015). This form of communication has the potential to provide probe-vehicle information. The communication protocol extracts data for each vehicle on the road and transmits it to an adaptive controller. The development of adaptive control systems under the assumption of the V2X environment has gained much attention in recent years.

Yao et al. (2020) proposed dynamic platoon dispersion for signal timing optimisation. The model predicts vehicle arrivals by using connected vehicle data. Then a signal strategy based on minimising average delay is utilised to set a green time duration. The rolling horizon methods required at least a 50% penetration rate to surpass other adaptive control algorithms.

Al Islam and Hajbabaie (2017) proposed a distributed-coordinated strategy for signal timing optimisation in a connected vehicle urban environment. The decision on terminating or continuing green times is made at the intersection level rather than the network level. Then, the intersections coordinate their decisions to achieve globally optimal decisions rather than

locally optimal solutions. The coordination method reduced the complexity of the traffic control problem. The simulation results indicate that the nominated algorithm can control the queue length and prevent spillback. Besides that, the suggested method decreased travel time between 17% and 48% and maximised throughput between 1% and 5% compared to actuated signals.

Feng et al. (2015) proposed a phase allocation algorithm to optimise a phase sequence and duration. The authors proposed an estimation of location and speed (EVLS) algorithm to construct a complete prediction arrival table. In the system, the algorithms are run with different objective functions, including minimising total vehicle delay and queue length. The simulation results indicated that the proposed strategy performs better than the fully actuated controller if the penetration rate exceeds 50%. Moreover, different objective functions result in different traffic signal timing behaviours. While minimising total vehicle delay generates a lower total vehicle delay, minimising queue length serves higher balanced signal phases but results in a higher total vehicle delay.

Guler et al. (2014) proposed a traffic signal control algorithm to minimise the total delay by optimising sequences of cars' departures from an intersection. The results showed that with 60% penetration rates, the system recorded the equivalent mark of average delay (i.e., 60%) and minimised the total number of experienced stops. In addition, the authors stated that no significant effect was achieved beyond a 60% penetration rate.

Goodall et al. (2013) developed a predictive microscopic simulation algorithm (PMSA) control to optimise an objective function to minimise delay only or a combination of delay, stops, and deceleration. Simulation results indicate that using the delay-only objective function performs better than the combination of delay, deceleration, and stops as the objective function. When the delay as the sole objective variable was used, the algorithm maintained or improved the performance compared to actuated timing signals at low- and mid-level traffic volumes and with a greater than 50% penetration rate of equipped vehicles. Moreover, the algorithm showed improved signal performance during unexpectedly high demand and the ability to automatically respond to year-to-year growth without retiming.

2.3.3 Review on Communication Channel Approach

Based on the literature review for the V2X environment, it is evident that such a communication channel boosts the performance of an adaptive controller, with the condition that the penetration rate should be at least 55% on average across different studies. When the penetration rate is low (below 30%), the adaptive controller loses its advantage and is constantly reported to perform worse than other controller systems. The prediction may yield significant variations, especially when the penetration rate is low (He et al., 2012). Tiaprasert et al. (2015) stated that queue estimation is not accurate in the absence of connected vehicles, and a higher penetration rate increases the accuracy of the estimation.

The costs of optimising existing infrastructures to accommodate V2X communication technologies are both expensive and slow. On-board computer processing units, towers, coverage, and bandwidth are a few examples that require alteration. Policy implications such as international communication protocols and communication infrastructure do not yet exist to successfully deploy vehicle agents (Raphael et al., 2015). Equipping vehicles with the necessary devices is still low compared to the total number of vehicles on the road. As the implementation of connected vehicles is expensive, the preference in practice is given to traffic light systems based on information from existing deployed infrastructure, i.e., traditional devices (Jin, 2018, Raphael et al., 2015). Table 2.2 presents adaptive controllers based on the communication protocol.

Table 2.2: Classification of adaptive traffic controllers based on communication channel

Communication Channel	Adaptive Traffic Control Study
Wired Communication (Road-to-Intersection)	Chu et al. (2019), Tan et al. (2019), Zeng et al. (2019), Deligkas et al. (2018), Casas (2017), Gao et al. (2017), Li et al. (2016), Raphael et al. (2015), Płaczek (2014), Smith et al. (2013)
Wireless Communication (Connected Vehicle)	Yao et al. (2020), Wang et al. (2019), Islam and Hajbabaie (2017), Feng et al. (2015), Tiaprasert et al. (2015), Khamis and Gomaa (2014), Guler et al. (2014), Gregoire et al. (2014), Wang et al. (2014), Goodall et al. (2013)

2.4 Classification Based on Traffic Control Strategy

The mission of the signal controller is to maximise traffic flow while considering various factors such as signal timing constraints, real-time strategies, and practical implementation (Eom and Kim, 2020). Several control strategies have been proposed to address the complexity of mitigation. The

intelligent logic mission is to employ timing parameters to achieve specific objectives (Ma et al., 2016).

The adaptive strategy executes signal logic criteria such as (i) green light extension to prolong the green phase, (ii) max out green phase split, or (iii) gap out to terminate a phase when the time interval between consecutive activations exceeds a predefined threshold (Eom and Kim, 2020). The core of signal optimisation is to achieve specific performance goals. Most of these objectives fall within two (2) broad classifications, including mobility and sustainability (Lee and Park, 2012). The mobility-based approach focuses on serving certain road user classes, but the sustainability-based approach targets extracted mobility features. The sustainability-based policy can have either a single objective or multiple objectives. The three (3) policy aspects are presented in the following sub-sections.

2.4.1 Road User and Vehicle Type

Some researchers considered a passenger car unit (pcu) without pedestrians to represent the traffic environment. Having a pedestrian impacts the minimum green light required to cross a street. Other researchers focused on priority strategy management based on private and public transit vehicles.

Zhou et al. (2017) proposed an active signal controller to reduce the delay of a bus rapid transit (BRT) mode of transport and maximise average passenger benefit. The controller uses the vehicle infrastructure integration

(VII) system to collect data. The VII channel provides precise data related to vehicles, including location and speed, to estimate the BRT travel time and arrival time. To test the algorithm, an isolated intersection was modelled for a BRT route in Jinan, China. Various traffic flow scenarios and a few signal priority strategies (green light insertion and green extension) were investigated in the study. The results showed that optimal signal priority strategies are related to green extension and red truncation. The findings showed that the proposed method improves the travel speed of BRT by 7.5% compared to existing signal controller systems. In addition, the system achieved an average of 19.35% reduction in passenger delay.

Zhang et al. (2017) proposed a traffic scheduling strategy based on quadratic programming (MIQP) for signal scheduling. The MIQP considers both pedestrians and vehicles in an urban traffic context. The study proposed a mathematical model comprising several logic constraints to describe the flow of pedestrians. The controller aimed to trade off the delay between pedestrians and vehicles and minimise the cost for both road users. The results indicated that the proposed controller handled light pedestrian assignment efficiently with a small number of junctions and prediction horizon.

Ma et al. (2014) presented a genetic algorithm-based heuristic algorithm for a multi-objective model. The model aimed to optimise the exclusive pedestrian phase (EPP) and two-way crossing (TWC) intersection. The method determines the optimal EPP to best accommodate delays in vehicular traffic and pedestrian traffic. The tests indicated that the proposed

model was much more effective in producing phase patterns and timing plans to address road users' needs than the SYNCHRO technique. Under low traffic volume, the TWC was suitable. In high-demand traffic flow, the EPP was suitable.

Wang et al. (2014) established a cooperative bus priority system (CBPS) based on connected vehicle technology. The system communicates with a public bus and a signal controller. The aim is to permit the bus to proceed unimpeded through an intersection. For testing, the authors deployed the CBPS system at an isolated intersection with two adjacent bus stops in Taicang City, Jiangsu Province, China. The results showed that the CBPS reduced the probability of the bus stopping from about 90% to 10%. As the bus experienced fewer stops, fuel consumption improved by nearly 27%.

Zeng et al. (2014) proposed a stochastic mixed-integer nonlinear program (SMINP) model to implement real-time TSP control. The SMINP prioritises bus movement by forecasting arrival times. The model considers the bus stop dwell time and delay caused by standing vehicle queues to estimate arrival time. The results showed that the SMINP yielded a 30% improvement in bus delay compared with RBC-TSP in a single-bus case, and the SMINP handled the bus priority much more effectively in a multiple-bus case.

Christofa et al. (2013) proposed a mixed-integer nonlinear program (MINLP) to minimise the total person delay at an intersection level by providing priority to transit vehicles based on their passenger occupancy.

He at al. (2012) introduced platoon-based arterial multi-modal signal control with online data (PAMSCOD) for multiple travel modes. The system identifies exiting queues and significant platoons approaching an intersection. The mixed-integer linear program (MILP) then determines the optimal future signal plan based on current controller status, platoon data, and any priority requests from transit buses.

Ekeila et al. (2009) presented a dynamic transit signal priority (DTSP) system comprised of an automatic vehicle location (VAL) detection system, a transit prediction model, and a priority strategy selection algorithm. The DTSP aims to minimise the delay of a transit vehicle while preventing negative impacts on street traffic. Whereas such a system can demonstrate that giving priority to certain road users can reduce delay, this comes at an inverting delay cost for non-priority users.

2.4.2 Classification based on a Single Objective

The objective of the signal controller is to provide safe and efficient passage of vehicles at the intersection (Wei et al., 2019). The reward function directs the priority of adaptive controllers to strategies for the following action of signal operation. Various traffic strategies are strategized for intersection

control. These optimisation factors are generally segmented into vehicle-based, time-based (value of time), headway and offset, and traffic flow.

2.4.2.1 Vehicle-based Objective

A feature extraction of individual vehicles, such as queue length, waiting time, throughput, and others, is deemed essential in determining signal cycle and control.

Queue length: Zaidi et al. (2016) designed an adaptive controller based on a back-pressure method. The scheduling algorithm weighs the pressure on a direction of travel based on queue length. The controller then activates a phase with the highest pressure release. Tiaprasert et al. (2015) introduced a queue-based adaptive controller using a discrete wavelet transform (DWT). The DWT enhances the consistency of queue estimation. Chin et al. (2011) developed a Q-learning traffic signal timing plan management (QLTSTM) system. The approach discretised a queue length into four (4) levels from low to high and a timing plan of 1 and 5 green phase seconds. The simulation results indicated that the proposed system reduced the average waiting time and queue length.

Traffic speed: Wang et al. (2019) developed a joint control model to optimise the speeds of vehicles. The coordinated system allows a platoon of vehicles to pass a series of signals with no stops or the least stop time. The proposed system led to savings in delay time and higher throughput.

Traffic Volume: Zheng and Liu (2017) proposed to optimise signal algorithms using estimated traffic volume. The estimation requires using GPS trajectory data from connected vehicles or navigation devices.

2.4.2.2 Value of Time

Becker (1965) was the first to introduce the concept of the value of time (VoT). Since then, the VoT parameter has been widely used to estimate wasted time at congested travelling corridors and to provide alternative and faster-tolled routes (highways) for road users (Bento et al., 2015). The application of VoT for intersection control appeared much later in the work of (Dresner and Stone, 2004). Dresner and Stone (2004) first introduced a reservation-based multi-agent system to manage intersections. Vehicles request a time slot (occupancy span) from the controller. Then the proposed system serves on a first-come, first-served basis. The authors mentioned that their reservation-based system outperformed conventional traffic lights.

Schepperle and Böhm (2008) created a valuation-aware mechanism algorithm. The proposed traffic controller takes into account the driver's VoT and allows concurrent use of intersections using an auction mechanism. The authors proposed two (2) mechanisms: Free Choice and Clocked. In Clocked, time slots are auctioned off, while in Free Choice, the winner selects preferred time slots from an interval. The authors concluded that Free Choice is always practical compared to Clocked, which is only adequate for higher demands and lower degrees of concurrency. Free Choice contributed around 38% of the

reduction in the average weighted waiting time. Their work assumes a fully connected environment.

Vasirani and Ossowski (2012) examined the combinatorial auction to induce changes to the reservation-based system proposed by Dresner and Stone (2004). In addition, the authors expanded their contribution by including multiple intersections. The authors studied various traffic densities impacting the intended amount to "pay" to use intersection-based delays experienced by drivers. The findings indicate that the combinatorial auction is effective for drivers willing to submit higher bids. The demand-response pricing policy led to the distribution of vehicles in the experimental network. Adapting the reserve price generates dynamic equilibrium as underutilised junctions become cheaper, and highly demanded junctions become more expensive. Hence, a homogeneous distribution of vehicles over the network can improve network resource utilisation and reduce travel time. The proposed mechanism assumes connected vehicle technology.

2.4.2.3 Headway and Offset

The signal optimisation can be viewed as an assignment that minimises total delay. Besides the cycle length and green split, an offset is crucial in defining a coordinated control plan. The offset and green split allow vehicle progression. The limitation of offsetting is that it requires coordination between intersections; in many intersections, this approach could face computation difficulties (Ma et al., 2016).

Ma et al. (2016) proposed a multi-stage stochastic adaptive program for coordinated signals. The method allowed extending green time to reduce residue queues by adjusting offset settings by switching between coordinated approaches and green time under oversaturated conditions.

Li et al. (2018) developed an adaptive coordinated controller for stochastic demand via phase clearance reliability (PCR). The method adjusted the signal offset to respond to stochastic demands on two (2) stage levels. The timing plan was developed to serve demand up to a certain PCR level. A queue clearance green was executed if a low reliability level was applied.

2.4.2.4 Traffic Flow

Younes and Boukerche (2016) proposed an intelligent traffic light controlling (ITLC) algorithm for the isolated intersection. Under ITLC, the flow with the maximum traffic density was scheduled to pass the intersection first. The results indicated that ITLC decreased the delay by 25% and increased the throughput by 30%.

Shaghaghi et al. (2017) introduced a VANET adaptive green traffic signal control (AGTSC-VC). The system divided signal control into two (2) levels: (i) VANET to assist in gathering traffic information, and (ii) traffic signal timing generation and traffic density assessment. The clustering algorithm is utilised to compute the density of the vehicle. Priority-based and density-based traffic signal timing methods improved the proposed approach's

performance better than the conventional method. The results demonstrated the advantage of AGTSC-VC to improve the accuracy of density estimation, decrease the waiting delay, boost the travel time of prioritised vehicles, and reduce gas emission rates.

Nafi and Khan (2012) presented an intelligent road traffic signalling system (IRTSS) using VANET communication. IRTSS aims to mitigate vehicle density at an intersection. The proposed IRTSS optimises fuel consumption by improving traffic flows. The proposed strategy significantly improved waiting time compared to a fixed cycle time control system. Nonetheless, IRTSS prioritised certain directions of travel if needed.

2.4.3 Multi-objective Controllers

The multi-objective combines two (2) or more traffic characteristics to strategize signal operation. Joo et al. (2020) proposed optimisation for traffic signal control by maximising the throughput and minimising the standard deviation of queue length. The proposed model was compared with RL controllers, including the work of the classical RL approach by Chin et al. (2011) and the approximated function-RL agent proposed by Liu (2017). The authors claimed to outperform these controllers regarding waiting time, average queue length, and standard deviation of queue lengths.

Raphael et al. (2015) suggested an intersection agent based on an auction system. The authors designed two (2) variations: saturation (SAT) and

saturation with queuing (SATQ). The bidding system extended the green phase for the winning travel approach. The decision is made based on the highest input. The results validated that SATQ is superior to other controller systems as it could perceive traffic state during auction execution. SATQ reduced travel time costs between 32% and 38% in comparison to SAT and fixed systems.

Brys (2014) studied the impact of control policy on state-action-reward-state-action (SARSA) algorithm. The objectives include delay and throughput. These policies were used individually and combined to guide the RL controller. The authors tested the performance at an isolated signal intersection. The findings indicated that the multi-objective approach achieved more excellent performance than a single objective. In addition, penalising the punishment of the controller by squaring the delay alone or combined with throughput yielded further improvement. The scalarization approach is a disadvantage, as the weights require careful tuning.

2.4.4 Review on Control Strategy

Two (2) principles of traffic lights include maximisation of the opportunity for cars to move without stopping and provision of flexible operation, which grants minimum delay. The challenge is to find an optimal signal configuration to achieve the control task. The presented controllers, based on green wave provisions, favour a particular class of road users. Such a logic technique is unsuitable for more comprehensive state implementation.

This class of strategies will negatively impact other road users due to greed and inclination. Other control methods, such as the VoT strategy, are challenging to weigh precisely. Utilising the VoT for junction control is fundamentally challenging as (i) coordination among intersections is needed and (ii) the VoT differs significantly in heterogeneous traffic environment (Deligkas et al., 2018), whereas the offset strategies require coordination among intersection controllers. Hence, the coordination mission becomes challenging, particularly for road network implementation where heterogeneous traffic volume and conflicting demands are present.

Subscribing to traffic states seems to be the right solution to mitigate intersections. After all, the mission of the controller is to regulate vehicle movements. However, feature extraction (vehicle-based) can potentially be deceptive for control optimisation problems, and essential information might be lost. The control agents respond to the environment and merely look into aspects of traffic flows. For instance, using queue length alone could lead to a spurious claim by assuming vehicles not in the queue are irrelevant. Similarly, using historical data such as flow rate yields a coarse approximation of the current state and ignores and abstracts away useful information (Genders and Razavi, 2016).

Furthermore, most of the cited works of literature intend to configure the incoming lane movements without considering the outbound movement. Only a handful of studies attempted to construct a policy compromising upstream and downstream flows. These studies are gaps in time and lack

researchers' support and investigation. Despite new technologies and innovative design mechanisms, there is little to no attention to innovative control strategies to maximise the potential of intelligent controllers. Table 2.3 summarises current studies based on the control policy method.

Table 2.3: Classification based on the control strategy for adaptive signal controllers

Control Policy			Adaptive Traffic Control Study
Mobility & Road User			Vilarinho et al. (2017), Zhou et al. (2017), Dai et al. (2016), Wang et al. (2014), Christofa et al. (2013), He at al. (2012), Ekeila et al. (2009)
Sustainability	**Single Objective**	**Queue length**	Li et al. (2016), Zaidi et al. (2016), Tiaprasert et al. (2015), Abdoos et al. (2013), Kari et al. (2014), Chin et al. (2011)
		Halting time and delay	Yen et al., (2020), Chu et al. (2019), Liang et al. (2019), Gao et al. (2017)
		Vehicle speed	Wang et al. (2019), Khamis and Gomaa (2014)
		Value-of-Time	Deligkas et al. (2018), Vasirani and Ossowski (2012), Schepperle and Böhm (2008), Dresner and Stone (2004)
		Traffic volume	Zheng and Liu (2017)
		Headway and offset	Ma et al. (2016), Li et al. (2018)
		Traffic flow and arrival	Yao et al. (2020), Pandit et al. (2013), Shen et al. (2018)
		Density	Younes and Boukerche (2016), Smith et al. (2013)
		Saturation	Raphael et al. (2015)
	Multi-Objective	**Vehicle position and speed**	Liang et al. (2019), Genders and Razavi (2016), Feng et al. (2015)
		Vehicle delay, stops and deceleration	Goodall et al. (2013)
		Throughput and delay	Brys (2014)
		Halting vehicles and waiting time	Zeng et al. (2019)
		Outflow rates and queue lengths of waiting and approaching vehicles	Thunig et al. (2019)
		Throughput and queue length	Joo et al. (2020)
		Saturation and queue length	Raphael et al. (2015)

2.5 Classification Based on Agent's Algorithm

Various algorithms are used to develop intelligent controllers, ranging from simple logistic regression to advanced machine-learning techniques. The logistic regression methods are typically less complex and do not require training. The regression algorithms act online to treat signal control problems.

Since the 1960s, the adaptive signal controllers have witnessed significant transformation due to the improvement in microprocessors and computing power, alonger with the advancement in machine learning (ML) and artificial intelligence (AI) methods. The adaptive control systems become more responsive to traffic demands. The ML methods are divided into three (3) classes: supervised, unsupervised, and reinforcement learning. All these techniques have been widely researched and addressed in adaptive logics.

The literature review for designing adaptive controller systems distinguishes between online and offline algorithms. The online algorithm utilises the learning during the deployment. This approach gives online learning an advantage in adaptation assignments (El-Ghazali, 2020). In this aspect, obtaining effectiveness becomes very costly, and extensive data will be required to achieve such supremacy. Whereas offline algorithms, as in ML methods, gather knowledge from training instances and generalise instance features to solve new instances. ML is an advantageous strategy for applications related to stochastic behaviour, such as traffic environment. The online algorithms are sufficient for slow changes in dynamics over time.

Nevertheless, a sudden change could cause online algorithms to fail to adapt. On the other hand, offline agents are much more flexible and can accommodate various traffic dynamics. Nevertheless, the offline training instances need to feature all instances of the environment.

2.5.1 Logistic Regression Controller

The self-organising traffic light (SOTL) and pressure techniques are very popular regression agents. The SOTL relies on agent interaction to communicate information and make decisions. Gershenson (2005) produced one of the initial applications for the SOTL. The author's work has shown that an agent-based controller can control traffic signals effectively. Cools et al. (2008) extended the work to test 12 intersections in the city of Brussels. In addition, Lämmer and Helbing (2008) tested the SOTL system on a hypothetical grid-like network.

McKenney and White (2013) developed a SOTL based on traffic volume count. One (1) agent controls each intersection. An agent receives input related to traffic volumes on edges leading to and from the intersection. The algorithm then assigns proportions of cycle length to each travel direction based on their total volume at the intersection. The length of each phase (minimum duration of 5 seconds) sums up to the cycle length. The agent's performance was investigated against a fixed controller for the simulated area of a 9x7 block section (50 signalised intersections) in the city of Ottawa, Canada. The results showed that the developed system had 7.36% higher

speed on the network level during the 11-hour simulation test. This performance drops to about 2% higher speed for the proposed STOL during the morning peak. The controlling agent requires communication with neighbouring intersections.

Wongpiromsarn et al. (2012) pioneered the back-pressure (BP) method for adaptive controller systems. For each junction, the algorithm computes a “pressure” associated with a traffic movement. The pressure is a weighted flow rate, computed by the difference between the number of vehicles at the inbound link and the number of vehicles at the outbound link. The local controller maximised the network's throughput by releasing the phase corresponding to the highest pressure. The testing environment is a network with 14 signalled intersections. The proposed system was tested against SCATS controller logic. The findings indicate that the queue length was reduced by a factor of three (3) using the BP controller compared to SCATS.

Thunig et al. (2019) extended the self-organized traffic light control algorithm developed by Lämmer and Helbing (2008) to the agent-based transport simulation MATSim. The optimisation strategy prioritised links based on predicted outflow rates and queue lengths of waiting and approaching vehicles. The control strategy minimised the waiting times and queue lengths at the intersection level. The algorithm was tested on a replica of 17 intersections in the city of Cottbus, Germany. The results showed that the adaptive controller successfully reduced delays and stabilised waiting queues compared to fixed and actuated controllers.

2.5.2 Supervised Learning Controller

Supervised learning is a form of machine learning taken from provided, labelled examples (a training set) by a knowledgeable external supervisor (Sutton and Barto, 2018). Fuzzy logic is an infamous supervised learning technique in adaptive controllers. Fuzzy logic maps between the system's input and output to quantify magnitudes (Madrigal et al., 2022). Fuzzy logic utilises a set of static rules to determine the preferred action for a traffic signal. While research shows that this approach is practical for small networks, the use of a static rule base means further updates to the system are required over time (Hnaif et al., 2019). In addition, it is challenging to generate an effective rule base for complex intersections containing a high number of possible phases

Madrigal et al. (2022) presented an adaptive traffic fuzzy logic controller. The system utilises flow rate to compute the cycle duration. The cycle duration is split into phases proportional to the arrived flow rates based on Webster's method. The authors tested the system on an isolated intersection calibrated on an actual traffic study. For comparison, five (5) systems were included: a fixed controller, a time-gap-based controller, a time-delay-based actuated controller, a fuzzy logic for green extension, and an adaptive fuzzy logic-based method with a modified Webster's formula. The authors report that the proposed controller achieved outstanding performance and preserved a fair balance between phases.

2.5.3 Unsupervised Learning Controller

Unsupervised learning is a machine learning paradigm where an agent intends to find structure hidden in unlabeled training data (Sutton and Barto, 2018). Evolutionary algorithms are a prevalent unsupervised method used in developing intelligent controllers. The evolutionary algorithm (EA) is an optimisation and search technique (Vikhar, 2016). The EA is suitable for applications where it is impossible to use heuristic solutions and survival for the fittest. EA suffers from some problems, like the fact that it needs lots of computational resources and is not assured to always give an optimal solution to a specific problem within a predictable timeframe. Genetic algorithms and heuristic algorithms are trendy in the literature of adaptive controllers. The Genetic algorithms choose the fittest priority to regulate vehicles at the traffic light intersection (Hnaif et al., 2019).

Putha et al. (2012) proposed a novel technique based on the Ant Colony Optimisation (ACO) algorithm to address oversaturated traffic conditions. The ACO system requires the departure rates, queue lengths, and arrival rates at the intersections to compute the objective function. The objective is to decide on green times to maximise network traffic throughput. The proposed model was tested against the genetic algorithm (GA) in two network configurations. Model I is a 4x5 1-way through movement network, and model II is a 4x4 grid network model with left-turn movement, different lane lengths, and the number of lanes compared to the model I. The findings indicate that the ACO presented significantly less variance among random trials. In addition, statistical analysis showed that the ACO yielded better

results for the same computational power, and the ACO is a good alternative for solving complicated networks compared to the GA system.

Hyper-heuristic technique is a methodology for selecting or generating solutions to multiple optimisation problems to generalise optimisation so systems can operate (Ahmed et al., (2018). There are two (2) levels: low-level candidate (heuristic) selection and move acceptance to decide on the generated solution. Galvan-Correa et al. (2020) designed a unique micro-artificial immune system algorithm (MAIS) to optimise the traffic light cycle. The proposed MAIS is tested in a model network (15 traffic lights) against simulated annealing (SA), genetic algorithm (GA), particle swarm optimization (PSO), and differential evolution (DE). The results reported that MAIS could achieve competitive performance compared to other systems. Under the MAIS system, vehicles experienced lower waiting times, between 7% and 24%, and reduced trip journeys ranged from 1% to 8%.

Gao et al. (2016) designed a discrete harmony search (DHS) algorithm to solve the signal light schedule. The system goal is to minimise the overall network delay using the traffic flow data. To evaluate the DHS system, the authors generated two (2) sets of traffic light scheduling from real-traffic data in Singapore. The first set has a time window of 30 seconds. The second set has a time window of 60 seconds. The grid model ranges from 3x3 to 10x10 grid networks. The DHS was compared with fixed cycle and standard harmony search (SH) traffic light systems. Based on the reported results, both swarm intelligent controllers, SH and DHS, have superior performance

compared to fixed controllers. On the other hand, DHS showed a better relative percentage deviation (RPD) than SH.

2.5.4 Reinforcement Learning Controller

The RL mimics the intelligent behaviour of a human being by interacting with and learning from the environment and taking corrective actions using a trial-and-error process (Nagabandi et al., 2018). The flexible orientation of the RL using a customised reward function allows the logic programme to choose which parameters to optimise (Mannion et al., 2015). The motivation behind using the RL for traffic signal control problems is its ability to accommodate the dynamics of the environment. It can assist in overlooking some challenging issues, such as defining flow rate (Khamis and Gomma, 2014). In addition, an RL agent can self-learn without external supervision and prior knowledge (Yau et al., 2017).

Khamis and Gomaa (2014) proposed multi-objective reinforcement learning based on a cooperative multi-agent framework. Using Bayesian interpretation, the controller agent simulates the driver's behaviour (acceleration/deceleration). The system aims to achieve multi-objectives in terms of waiting time, trip time, flow rate, fuel consumption, and others. The performance was evaluated against two (2) adaptive controllers: self-organising traffic lights (SOTL) and genetic algorithm (GA). The test models included (i) a network of nine (9) signalled intersections and (ii) a city centre network replica of 22 traffic signals. Two (2) traffic situations were modelled

(congested and free scenarios). The tests showed that the proposed traffic system outperformed the single objective controller in terms of lowering waiting time, reducing the number of stops, minimising queue length, and improving the throughput. In addition, the authors recommended using a scalar-based reward design to boost the performance indices. The system requires full access (observation) to environmental attributes.

Mannion et al. (2015) proposed parallel reinforcement learning (PRL) for a multi-agent system. The PRL devised multiple agents to learn concurrently on a problem and share experiences to expedite learning and reduce convergence time. The PRL consists of two agent types: (i) a master agent and (ii) slave agents. The slave agents share experiences with a pool (Global Q matrix). The master agent can use experience from the pool or execute its own experience. The testing was carried out in a simple isolated junction with two (2) layout configurations, including two (2) approaches and a T-junction. The test set comprised PRL learning agent variations (2, 3, and 4). The results showed that the performance of the proposed PRL system increased with the number of parallel learners.

Pham et al. (2013) presented a tile coding method to approximate the value function SARSA agent. The agent has three (3) input states, including lapsed time for the last light change, number of seconds for the second-to-last light change, and total waiting time at the intersection level. The reward function dealt with waiting time. The controller had a small time duration of 2 seconds for every action. The controller either maintained the status quo or

changed to another phase. The controller was tested on a 2x2 grid network. In a low traffic scenario, the SARSA controller surpassed other systems in delay time. In a high-traffic situation, the SARSA performed better than comparative systems in delay and throughput. The authors explained the superiority of the RL system to outperform the coordination DCEE framework as the first is capable of relating state information to the problem (learning environment).

2.5.5 Review of Algorithm Technique

Traffic control is defined as an interactive problem. The interactive problem is a category of environment where the resembled behaviour for every situation cannot be comprehended entirely (Sutton and Barto, 2018). There are various algorithms used for developing adaptive controllers. Some of these methods require much more complex designs than others and are prerequisites to achieving efficient performance. For example, nearest neighbours and self-organising traffic lights (SOTL) require coordination with neighbouring intersections to achieve optimal operation.

Other systems are bound by expert knowledge to define the appropriate rules, as in fuzzy logic. Knowledge reliance limits the goal of developing a self-sufficient controller to deal with unpredictable situations and unknown traffic attributes. Supervised learning requires big data or constant experimentation to achieve sustenance. Unsupervised learning can be very costly for dynamic problems such as traffic signal controllers. The natural driving environment changes continuously. On the other hand, reinforcement

learning (RL) can be utilised to approximate the unknown, but the MDP assumption restricts its efficiency to exposed training data. Table 2.4 presents the design technique and corresponding limitations.

Table 2.4: Overview of main limitation in considering design technique

System Design	Suitable with	Studies
Logistic Regression	Slow dynamics	Varaiya (2019), Thunig et al. (2019) Hao et al. (2018), Younes and Boukerche (2016), Zaidi et al. (2016), McKenney and White (2013), Joo et al. (2020), Li et al. (2018), Wongpiromsarn et al. (2012)
Supervised Learning	Expert knowledge	Madrigal et al. (2022), Yao et al. (2020), Jiang et al. (2021), Tunc et al. (2021), Ali et al. (2021), Bi et al. (2014)
Unsupervised Learning	Constant environment	Galvan-Correa et al. (2020), Hnaif et al (2019), Darmoul et al. (2017), Gao et al. (2016), Ma et al. (2014), Putha et al. (2012)
Reinforcement Learning	MDP and Large data set for training	Wan and Hwang (2018), Gao et al. (2017), Genders and Razavi (2016), Mannion et al. (2015), Khamis and Gomaa (2014), Pham et al. (2013), Chin et al. (2011), Priemer and Friedrich (2009)

2.6 Review of Application of Adaptive Traffic Controller and Challenges

The literature review has over 60 studies on adaptive controllers. These intelligent systems were segmented into four (4) categories based on system design, control strategy, communication channel, and control algorithm. The common drawback refers to a persistent issue requiring immediate action to advance research in this field.

2.6.1 Environmental Settings

Issue 1: Traffic simulation is a prerequisite for testing intelligent traffic control systems. The modelling application aims to reproduce human decision-making and behaviour, capturing the level of detail required for a particular objective (Bazzan and Klügl, 2014). Single intersections are very popular in evaluating adaptive controllers. Nearly 50% of revised works in adaptive controllers utilised isolated environments, as in Figure 2.2. Despite reporting superior performance in isolated intersections, whether the proposed controllers are applicable for mitigating network-level operations is unknown. **The proposed control methods do not provide details on how mitigation is achieved beyond the isolated intersection.**

On the other hand, the grid network represents a traffic situation where no conflicting movements are faced. Such a testing configuration does not reflect a challenge for adaptive systems. In addition, fixed controller systems are efficient for this type of network (Gordon and Tighe, 2005). Only three (3) in 10 studies considered arterial networks, and a smaller proportion integrated real-world scenarios.

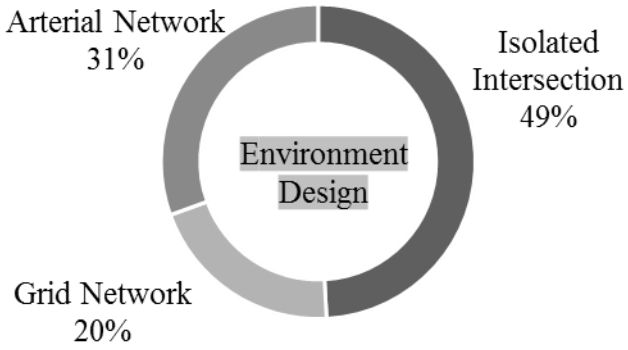


Figure 2.2: Environment setting for intelligent controller development

Issue 2: The training and evaluation are commonly performed in ideal simulated traffic scenarios. Figure 2.3 shows that the majority (75%) of adaptive controllers are tested in hypothetical testing sets. These traffic scenarios are not related to real-world applications. The traffic demand is hypothetical, and the route choice behaviour is simplified. The simplest methodology for creating several assumptions, such as constant traffic flow, fixed headways and gaps, default values of saturation flow, uniformity of vehicle class, and others, makes the proposed approaches rigid and impractical for real-world class. The current in-practice adaptive systems such as TRANSYT (Robertson, 1969), Split Cycle and Offset Optimisation (SCOOT) (Hunt, 1982), and Sydney Coordinated Adaptive Traffic Systems (SCATS) (Lowrie, 1982) suffer from presuming default saturation flow. Such a definition undermines traffic flow patterns, and on large city corridors, these systems may fail (Bazzan and Klügl, 2014). **Therefore, the effectiveness of adaptive signal agents in a real-world application is hard to prove (Gong et al., 2019). Robust and more flexible systems for environmental dynamics need to be examined.**

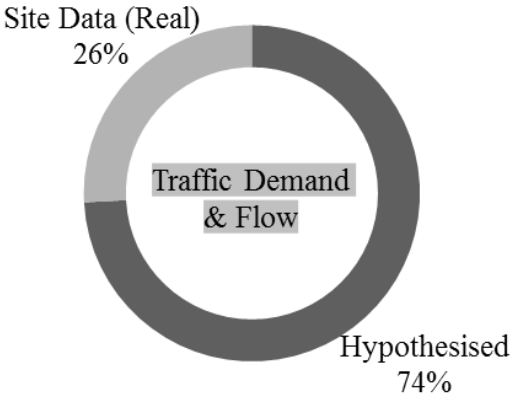


Figure 2.3: Traffic flow and demand for an evaluative environment

2.6.2 Communication Protocol

Issue 3: To capture environmental states, researchers utilise two (2) methods: real-time traffic feedback and approximation functions. The latter method simplifies the traffic representation, but the hypothetical context (issue 2) restricts their legitimacy integration. The approximate functions do not account for mixed transport modes and consider traffic to exhibit homogenous features. Some approaches suggested integrating approximation functions to improve the online algorithms. The controller with approximate capability could perceive the near future state and decide on suitable action. **It is unknown whether these approximate techniques will eventually capture global mitigation**. The rolling horizon is short (every signal phase). In addition, **what is the sensitivity of these approximate methods to error cases? The error occurs if the approximate function fails to produce information, leading to an inappropriate phase split. This case raises another question: How will the agent recover from such a decision error?**

Issue 4: Four (4) out of 10 studies did not verify the communication protocol. as shown in Figure 2.4. Not verifying the communication channel indicates that the authors assumed fully observed driving corridors. This assumption is naïve, as current detection devices monitor traffic within defined boundaries. In addition, the current infrastructure is not equipped to accommodate connected vehicle technologies. If this is the case, then the penetration rate dilemma is also applicable to this category of studies. It can be stated that 70% of proposed adaptive methods' efficiency relies on accurate environmental representation. Despite the innovative methods, whether these systems can function in a **restricted data feed environment remains unanswered**. Overall, the ability of the proposed controllers to function in real mixed-mode traffic using the existing detection devices has not been extensively explored.

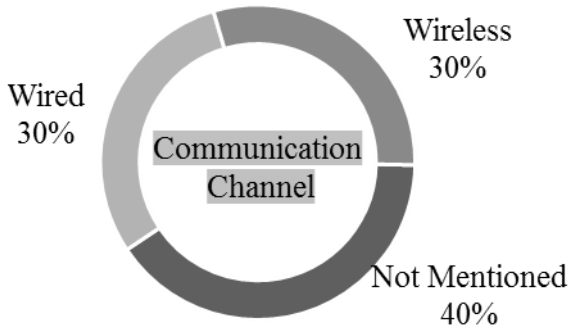


Figure 2.4: Communication channel protocol in adaptive control studies

2.6.3 Split Optimisation

Issue 5: Agents are the traffic signals, but a priori determination of the signal plan is commonly cited in control systems. **Restricting an intelligent controller to decide on a signal plan assignment contradicts the adaptation mission.** Traffic is a highly dynamic process, and signal plans should not be determined in advance. Figure 2.5 shows that 82% of intelligent controllers execute a restricted role of phase allocation or phase duration. Only less than 10% of the studies developed free-cycle controllers.

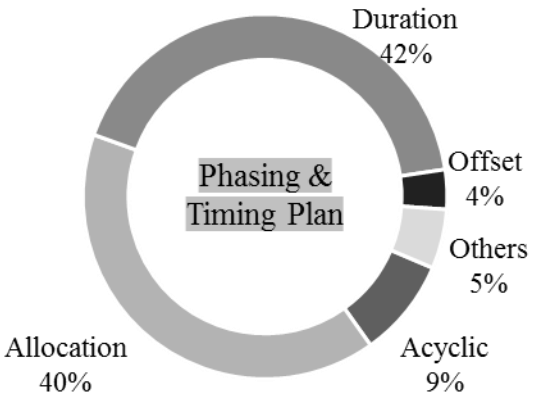


Figure 2.5: Technique used in logic's function

2.6.4 Control Optimisation

Issue 6: There are three (3) popular designs: (i) organising control agents in a hierarchical structure, (ii) letting agents decide on local control and coordinate with neighbours, and (ii) solo action agents. The first centralised decision relies heavily on communication. Though the hierarchical system is

advantageous in resolving conflict, it is a pure communication-based approach. The data transmission protocol is expensive and futuristic technology (V2X), and reported experiments showed that the inaccurate data exchange cripples the signal operation. **So until the road environment becomes fully connected, centralised systems are not practically available to solve the existing traffic control problem**.

Issue 7: The coordinating action systems have a simpler communication channel. An agent can only communicate with its immediate neighbours to synchronise action. On the other hand, the horizontal system has a potential drawback in resolving conflicts. The synchronised action is a form of green wave that allows a platoon of vehicles to cross several intersections. **The assumption that the traffic flow is unanimous and non-conflict will eventually restrict the applicability of progressive traffic signals to work efficiently in real intersection networks and opposing traffic flows**.

Based on Figure 2.5, multi-agent systems (centred and coordinated decisions) dominated the literate studies at 65%.

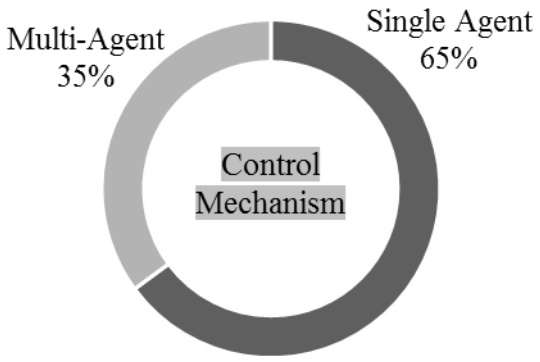


Figure 2.6: Design mechanism for adaptive controller

Table 2.5 presents a summary of adaptive control studies in this literature review.

Table 2.5: Summary of research studies in adaptive traffic control

No.	Author	Algorithm Technique	Time Plan and Signal Control	Traffic Control Strategy	Optimisation Mechanism*	Communication Channel	Simulation Environment
1	Genders and Razavi (2016)	Reinforcement learning	Phase duration	Cumulative delay	Local	Wired	Isolated Intersection (Hypothetical)
2	Gao et al. (2017)	Reinforcement learning	Phase duration	Waiting time	Local	Wired	Isolated Intersection (Hypothetical)
3	Li et al. (2016)	Reinforcement learning	Phase duration	Absolute value difference for queue length	Local	NA	Isolated Intersection (Hypothetical)
4	Casas (2017)	Reinforcement learning	Phase duration	Factor for speed score scaled by a discount factor and vehicle counts	Local	Wired	Isolated Intersection (Hypothetical), 3x2 Hypothesised Grid Network (Hypothetical) & 43 Junctions Network (Real Network)
5	Wan and Hwang (2018)	Reinforcement learning	Phase allocation and duration	Cumulative time delay between 2 actions	Local	NA	Isolated Intersection (Hypothetical)
6	Liang et al. (2019)	Reinforcement learning	Phase duration	Cumulative waiting time	Local	NA	Isolated Intersection (Hypothetical)
7	Zeng et al. (2019)	Reinforcement learning	Phase duration	Linear combinations of normalised halting vehicles, waiting time for vehicles at intersection, and scalar quantity punishment	Local	NA	Isolated Intersection (Hypothetical)
8	Chu et al. (2019)	Reinforcement learning	Inverse of the waiting time	Waiting time	Local	Wired	Isolated Intersection (Hypothetical)
9	Wang et. al. (2019)	Reinforcement learning	Phase allocation	Throughput factor and a waiting time factor with trade-off coefficients	Local	Wired	Isolated Intersection (Hypothetical)
10	Yen et al. (2020)	Reinforcement learning	Phase allocation	Ratio of throughput to average end-to-end delay	Global	Wireless	3x3 Grid Matrix (Hypothetical)
11	Hao et al. (2018)	Logistic regression	Phase allocation	Minimum delay	Local	NA	4x4 Manhattan Grid (Hypothetical)
12	Płaczek (2014)	Logistic regression	Phase duration	Minimum delay	Local	Wired	4x4 Grid Network (Hypothetical)
13	Varaiya (2019)	Logistic regression	Phase allocation	Maximum throughput	Local	NA	Na
14	McKenney and White (2013)	Logistic regression	Phase duration	Maximum throughput	Local	NA	9x7 Block (50 Signalised Intersection) (Real)
15	Younes and Boukerche (2016)	Logistic regression	Phase allocation	Maximum density	Local	Wireless	Isolated Intersection (Hypothetical)
16	Joo et al. (2020)	Logistic regression	Phase distribution	Maximising throughput and minimising deviation of queue length	Local	NA	Isolated Intersection (Hypothetical)
17	Chin et al. (2011)	Reinforcement learning	Phase duration	Minimum queue length	Local	NA	Isolated Intersection (Hypothetical)
18	He at al. (2012)	Logistic regression	Phase duration	Prioritise transit buses	Global	NA	Isolated Intersection
19	Christofa et al. (2013)	Logistic regression	Phase allocation	Minimise the total person delay	Local	Wired	Isolated Intersection
20	Ekeila et al. (2009)	Logistic regression	Phase allocation	Minimise delay of transit vehicle	Local	Wired	Isolated Intersection

No.	Author	Algorithm Technique	Time Plan and Signal Control	Traffic Control Strategy	Optimisation Mechanism*	Communication Channel	Simulation Environment
21	Hanif et al (2019)	Unsupervised learning	Phase duration	Mean speed, traffic density and traffic flow	Local	Wired	4 Intersections (real)
22	Putha et al. (2012)	Unsupervised learning	Phase duration	Maximising throughput	Local	NA	4x5 (1-way through) Network (hypothetical), and 4x4 grid network (real)
23	Zaidi et al. (2016)	Logistic regression	Phase allocation	Minimising queue length	Local	NA	24 Intersections (real)
24	Wongpiromsarn et al. (2012)	Logistic regression	Phase allocation	Maximising throughput	Local	NA	Isolated intersection (real) and 14 signals (real)
25	Smith et al. (2013)	Logistic regression	Phase allocation	Maximise traffic flow	Global	Wired	9 Intersections (real)
26	Deligkas et al. (2018)	Logistic regression	Phase duration	Minimising total waiting time	Local	Wired	Isolated intersection
27	Raphael et al. (2015)	Logistic regression	Phase duration	Minimising total waiting time	Global	Wired	Grid network (25 signals)
28	Vasirani and Ossowski (2012)	Logistic regression	Phase allocation	Minimise vehicle delay	Local	Wireless	Isolated intersection
29	Zhou et al. (2017)	Logistic regression	Phase allocation and duration	Minimise delay of BRT	Global	Wired	Isolated intersection
30	Wang et al. (2014)	Logistic regression	Phase allocation	Green phase for bus	Local	Wireless	Isolated intersection
31	Gregoire et al. (2014)	Logistic regression	Phase allocation	Minimise queue blockage	Local	Wireless	Grid network (64 signals)
32	Galvan-Correa et al. (2020)	Unsupervised learning	Phase allocation	Minimise waiting time, and maximising speed	Global	NA	Network (15 traffic lights) (real)
33	Gao et al. (2016)	Unsupervised learning	Phase allocation	Minimise waiting time	Global	NA	Grid network (9 to 100 intersections)
34	Madrigal et al. (2022)	Supervised learning	Cycle duration (phase split and timing)	Reduce queue length, waiting time, and density	Local	Wired	Isolated intersection
35	Li et al. (2018)	Logistic regression	Offset timing	Minimise delay	Global	NA	3 intersections
36	Ma et al. (2016)	Logistic regression	Offset timing	Minimise delay and overflow	Global	NA	3 intersections
37	Kővári et al. (2021)	Reinforcement learning	Phase duration	Occupancy and empty phase punishment factor	Local	Wired	Isolated intersection
38	Tan et al. (2019)	Reinforcement learning	Phase allocation	Balancing queue length and moving vehicles	Global	Wired	4x3 and 4x6 grid networks
39	Khamis and Gomaa (2014)	Reinforcement learning	Phase allocation	Waiting time, trip time, flow rate, fuel consumption, flow rate, green waves, and accident avoidance	Global	Wireless	Grid network (9 signals & 22 signals)
40	Mannion et al. (2015)	Reinforcement learning	Phase duration	Cumulative waiting time	Global	NA	Isolated intersection
41	Tiaprasert et al. (2015)	Logistic regression	Phase duration	Minimise queue length	Local	Wireless	Isolated intersection
42	Wang et al. (2019)	Logistic regression	Phase duration	Minimise delay and increase speed	Local	Wireless	Arterial network (5 intersections)

No.	Author	Algorithm Technique	Time Plan and Signal Control	Traffic Control Strategy	Optimisation Mechanism*	Communication Channel	Simulation Environment
43	Yao et al. (2020)	Supervised learning	Phase duration	Minimise vehicle delay	Local	Wireless	Isolated intersection
44	Darmoul et al. (2017)	Unsupervised learning	Phase allocation and duration	Traffic fluidity (improvement of queue length between neighbouring agents)	Global	NA	3 signalised intersections
45	Goodall et al. (2013)	Logistic regression	Phase allocation	Delay, stops and deceleration	Local	Wireless	4 signalised intersections (real)
46	Gong et al. (2019)	Reinforcement learning	Phase allocation and duration	Minimise cumulative waiting time	Global	Wireless	8 intersections (real)
47	Pandit et al. (2013)	Logistic regression	Phase allocation	Minimise waiting time	Local	Wireless	Isolated intersection
48	Rasheed et al. (2020)	Reinforcement learning	Phase allocation	Lowering waiting time and increasing throughput	Global	NA	7 signalised intersection (real) and 3x3 grid network (hypothetical)
49	Kari et al. (2014)	Logistic regression	Phase duration	Travel time and fuel consumption	Global	Wireless	Isolated intersection
50	Liu et al. (2017)	Reinforcement learning	Phase allocation and duration	Reduce waiting time	Global	Wireless	96 intersections (real)
51	Priemer and Friedrich (2009)	Reinforcement learning	Phase allocation	Minimise queue length	Global	Wireless	3x3 grid matrix
52	Wang et al. (2020)	Logistic regression	Phase duration	Minimise delay and increase throughput	Global	Wireless	5 signalised intersections (arterial) (real)
53	Zeng et al. (2014)	Logistic regression	Phase duration	Minimum delay for bus	Local	NA	Various grid networks (9,16,25,49,64,100,225 signals)
54	Ma et al. (2014)	Unsupervised learning	Phase duration	Minimum delay for road users (vehicles & pedestrians)	Local	NA	Isolated intersection
55	Byrs et al. (2014)	Reinforcement learning	Phase allocation	Minimise delay and increase throughput	Local	Wireless	Isolated intersection (hypothetical)
56	Thunig et al. (2019)	Logistic regression	Phase duration	Reduce waiting time and queue length	Local	Wired	17 intersections (real network)
57	Pham et al. (2013)	Reinforcement learning	Phase allocation	Waiting time at intersection level	Global	NA	2x2 grid network (hypothetical)
58	Al Islam and Hajbabaie (2017)	Logistic regression	Phase allocation and duration	Maximise throughput and minimise queue length	Global	Wired	2 intersections & grid network (9 intersections)-hypothetical
59	Tunc et al. (2021)	Supervised learning	Phase duration	Minimise waiting time and queue length	Local	NA	Isolated intersection
60	Jiang et al. (2021)	Supervised learning	Phase duration	Reduce delay	Local	NA	Isolated intersection (real)
61	Ali et al. (2021)	Supervised learning	Phase duration	Reduce waiting, travel times and increase speed	Local	Wired	Isolated intersection (real)

*Local refers to single agent design. Global refers to a multi-agent design

2.7 Deep Reinforcement Learning as Adaptive Traffic Control System

The RL-based method is a promising self-learning technique that is not bound by expert knowledge. The evolution of RL-based adaptive controllers can be categorised based on two (2) frameworks, including Markov decision process (MDP) and partial observable MDP or (PO-MDP).

The MDP considers a full state representation and environment are known. The earliest MDP studies of adaptive controllers are found in Thorpe (1997) and Abdulhai et al. (2003). Both of these studies expeditiously caused a state-space representation complication as their agents needed to map every possible arrangement of states (Khamis and Gomma, 2014). The drawback of the MDP systems is the computational requirement, which grows exponentially with road links with multiple traffic junctions. In fact, it is nearly impossible to have a fully observed environment specifically for traffic conditions.

In order to reduce the complexity and computational cost, a partially observable state environment is considered using the approximation technique. The PO-MDP is first introduced by Prashanth and Bhatnagar (2011). The authors introduced a function approximation for the Q-learning algorithm to develop a traffic control signal (QTLC-FA). The results indicated that QTLC-FA outperformed a Q-learning (MDP) algorithm and a fixed traffic signal controller in various test scenarios. Abdoos et al. (2013) proposed a linear function approximation for a Q-learning agent. The control algorithm

outperformed standard Q-learning in reducing the delay time across the network. Yin et al. (2014) proposed optimisation for a traffic controller based on an approximate dynamic programming (ADP) approach. The results indicated that the ADP outperformed existing controller strategies, including fixed-time and actuated controls, under high traffic loads.

On the other hand, the PO-MDP is similar to the MDP in that it considers a stationary traffic environment. Hence, in high dimensional state space (as in a realistic driving environment), the function approximation cannot efficiently learn the attributes of the environment (Haydari and Yilmaz, 2020). To overcome this challenge, a deep learning (DL) structure based on function approximation is appropriate for handling feature learning. Feature learning is achieved by extracting useful patterns from data and forming feature maps (Sutton and Barto, 2018). The DL herein refers to using multiple layers of an artificial neural network to learn feature mapping.

Sections 2.7.1 to 2.7.6 present prominent studies in deep reinforcement learning (DRL) control systems based on their neural structure, followed by challenges and drawbacks in Section 2.7.8.

2.7.1 Deep Q-Learning Controllers (DQL)

2.7.1.1 Convolution Neural Network for Adaptive Controller - Genders and Razavi (2016)

Control Agent: Genders and Razavi (2016) were the first to propose a convolutional neural network (DQTSCA). The action-value model used discrete traffic state encoding (DTSE) to monitor the environment's states. The DTSE comprised three (3) vectors: vehicle presence, vehicle speed, and current signal phase. The DTSE discretized inputs within a cell length of lane. The action space is pre-configured phase definitions. There are four (4) possible actions the DQTSCA could choose from. The control strategy is based on changes in cumulative vehicle delay between subsequent actions. The DQTSCA has two (2) hidden convolution layers, two (2) fully connected layers, and one (1) output layer (action call).

Experimental Design: The authors micro-modelled a 4-way isolated intersection with three controller movement definitions (left, through, and right turns). The observed lane length is 75 metres, which was divided into 12 DTSE (1 cell = 5 metres). The traffic flows used probability distributions ranging from 0 to 150veh/hr and from 250 to 450veh/hr. The phasing time is fixed at 7 seconds (2 seconds green and 5 seconds transition).

For comparison, a shallow neural network (TSCA) was designed. The TSCA agent has one (1) hidden layer with 64 neurons using a sigmoid activation functions and four (4) neurons with a linear activation function for

the output layer. The TSCA agent has two (2) states input: queue length and signal phase. DQTSCA and TSCA agents have similar reward and action attributes and were trained using the same number of epochs and gradient descent algorithm.

Results: To validate the proposed system, the authors compared data from each agent's last 100 training epochs (>93% exploitative action). The results indicated that the proposed DQTSCA agent reduces cumulative delay by 83%, average queue length by 66%, and travel time by 20% compared with STSCA. The authors explained the significant achievement in cumulative delay as the DQTSCA agent interacts with the cumulative delay parameter as a reward function. On the other hand, there is a negligible throughput difference between both agents. This attribute is an indication of the DQTSCA agent's policy fairness. A fair policy is expected to ensure that all vehicles are given equal priority to traverse the intersection. Although the authors compared two (2) RL agents, there is a conflict in considering different state representations for each system.

2.7.1.2 Deep Q-Learning Network with Experience and Target Network - Gao et al. (2017)

Control Agent: Gao et al. (2017) extended the work of Genders and Razavi (2016) and introduced a target network for the deep convolution neural agent (DQN). The experience replay and target network mechanism improved the agent's stability. Stability is defined as the ability of the trained agent to

make steady control decisions without diverging to bad action policies or oscillating between good and bad action policies. The state is based on DTSE (vehicle position, speed, and signal status). The two (2) vehicle state vectors were fed to convolution layers. The signal status input has a size of two (2) to indicate the location of the green phase. These input layers are concatenated into two (2) fully connected layers, followed by a fully connected output layer (action). For an action decision, there are two (2) action spaces for the agent's selection. The reward policy is to reduce the cumulative waiting time at the intersection.

Experimental Design: The authors modelled a 4-way signalled intersection with three turning movements (left, through, and right) to test the DQN. The lane length is 500 metres, and the observed length is 160 metres, with a cell length of 8 metres for the DTSE. Vehicle length is 5 metres with a 2.50 metres minimum gap between vehicles. High and low traffic flow demands were designed for the micro-model using the Bernoulli process. The traffic light has signal durations of 10 seconds for green and 6 seconds for yellow phases.

The authors utilised the longest-queue-first algorithm (LQFA) and fixed control algorithm to evaluate performance measures for the testing stage.

Results: Overall, the simulation showed that the developed controller reduces vehicle delay by 47% and 86% compared with LQFA and fixed time controllers, respectively. It was also observed that when traffic demand

increases, the average delay by the proposed DQN increases as well. In two out of the four (4) intersection routes, the proposed DQN led to a higher vehicle delay when the rate control parameter $p>0.80$ compared to LQFA. A similar observation was made for one of the routes versus the fixed controller. The authors have not clarified these findings. Moreover, the extension of the stability of the mentioned controller was not examined extensively. The testing and training sets had similar flow hierarchies (major and minor flows). Ideally, the trained deep reinforcement learning algorithm gains an understanding of environmental dynamics. Hence, if a testing set closely reflects these dynamics, the trained agent will achieve a favourable performance.

2.7.1.3 Deep Q-learning Neural (DQN) Network using Dynamic Discount- Wan and Hwang (2018)

Agent Controller: Wan and Hwang (2018) proposed a deep Q-learning neural (DQN) network. The authors proposed a dynamic discount factor to prevent a biased estimation of the action-value function. The agent used an experience reply and target network similar to Gao et al. (2017). DTSE enclosed vehicle positions and signal phase states. The control policy minimised the accumulated time delay between two subsequent actions. The phase action is chosen from a predefined phase signal plan.

Experimental Design: The authors constructed a 4-way isolated intersection. There are three (3) directions of travel (left, through, and right

turns). The arrival is random, but the traffic demand is fixed. The amber light is fixed to 3 seconds, whereas the green phase corresponds to five (5) seconds. The cycle phase is 180 seconds.

For comparison, the authors developed a deep Q-learning logic with one (1) state input (vehicle position) (DQN-1) and a fixed timing plan. For testing, the authors proposed two (2) scenarios, including (i) an unsaturated scenario with a traffic flow of 800veh/hr for each direction and (ii) an oversaturated scenario with a traffic flow of 2,000veh/hr for each north and west directions and 100veh/hr for the east and south directions.

Results: The findings indicated that the proposed trained agent outperformed the pre-timed signal plan for reducing total system delay by 20% and 15% for unsaturated and oversaturated conditions, respectively. In addition, the proposed controller improved throughput by 17% in an oversaturated condition compared to fixed timing. The results showed that the DQN-1 had worse performance than the fixed controller. The authors related the poor performance of the DQN-1 to a lack of the discount factor. However, the experiments were not conclusive enough to strongly support their justification, and there is a good chance that the DQN-1 had not converged to the optimal policy during training.

2.7.1.4 Mixed Deep Q-Network (MQN)- Zeng et al. (2019)

Control Agent: Zeng et al. (2019) introduced a unique deep Q-learning algorithm consisting of two (2) branches: a softmax classification and a Q-value network. The mixed Q-Network (MQN) agent was integrated with a memory palace (MP). The MP's function is to induce prior experience knowledge to the MQN to enhance learning. The state input was represented by DTSE, similar to Genders and Razavi (2016). The MQN had a multi-objective reward comprising five (5) metrics. The reward metrics are (i) counts of vehicles passing the stop line, (ii) counts of halting vehicles in the green phase direction, (iii) a phase skip punishment, (iv) a normalised halting factor of vehicles at the intersection level, and (v) total waiting time of vehicles around the intersection. There is four (4) action space for the MQN controller.

Experimental Design: The authors tested an isolated cross-intersection. Each approach direction has an inner lane for a left turn, two (2) middle through lanes, and an outer shared lane for right and through movements. The observed length of the lane is 120 metres, and the vehicle length (cell length) is five (5) metres, with two and a half (2.5) metres as a minimal safe gap. The yellow phase is fixed to four (4) seconds, while the minimum green phase is six (6) seconds and is capped at 60 seconds with two (2) seconds of incremental extension at each time step.

In order to evaluate the MDQN, a fixed controller and deep Q-learning with experience reply (DQN) were developed. The test bed is a three-hour simulation with two (2) classes of traffic demand (low and high). The low traffic demand is half the high demand flow. A uniform increment to simulate varied traffic flow was used for the simulation model. The left turn had 50 to 200veh/hr, the through movement was 150 to 600veh/hr, and the right turn had 100 to 400 veh/hr.

Results: Based on the results, both reinforcement learning agents surpassed the performance of the fixed signal system. On the other hand, the performance of the MDQN is not completely monotonous compared to a DQN. The MDQN outperformed the DQN in average reward value. Nevertheless, DQN has outperformed MDQN in average delay, average queue length, and average travel time. The authors reported that these differences are insignificant between both deep learning techniques. The contribution of the MP method to enhancing the MDQN is not detailed in the study.

2.7.1.5 Deep Q-learning Network Controller- Tan et al. (2019)

Tan et al. (2019) tried to solve the action space challenge associated with controlling a large-scale network by introducing a cooperative deep reinforcement learning (Coder) framework. The system breaks down RL tasks into a number of sub-problems with relatively easy RL goals. This design was achieved by dividing the network region into sub-regions. Each agent learns to achieve a solution. Then, a centralised global agent aggregates these solutions

and forms the final Q-function for the traffic grid. Overall, the agent intends to balance the costs of waiting in a queue and moving vehicles. The agent chooses from a pre-defined phase list to meet the control policy requirement. The Coder includes two (2) dense hidden layers for inputs.

Experimental Design: The authors constructed two (2) hypothetical grid networks (B and C) to test the Coder system. Network B is a 4x3 intersection network and was portioned into two (2) 2x3 sub-regions. The network had evenly distributed 904 vehicles. Network C is a 4x6 intersection network and was portioned into four (4) 2x3 sub-regions. The network had evenly distributed 1,344 vehicles. The phase time for Coder's action was restricted to five (5) seconds, and it can be extended or terminated at the end of the time interval. The distance between each adjacent intersection is 150 metres.

The authors compared the proposed Coder algorithm for the testing stage against fixed, random, linear Q-learning, and R-DRL.

Results: According to the observation, the Coder outperformed all the other methods in terms of waiting time and waiting for queue length. In network B, the Coder achieved a lower average queue length of 33% and a shorter waiting time of 71%. In network C, the Coder registered a lower average queue length of 28% and a shorter delay of 63%. The authors attribute the drop in performance when comparing networks B and C to the number of sub-regions (network B= 2, network C= 4). As the number of sub-regions

increases, the agent has to coordinate more local agents and search for improved local optimisation solutions in a much larger discrete action space. Thereof, the authors recommended restricting the sub-region size to ≤4. This limitation in learning might not be optimal in a stochastic traffic environment where the variation of traffic flow is evident and leads to complexity. In addition, the Coder requires a total observed environment and assumes the environment is in MDP states.

2.7.1.6 Capacity as Control Strategy for Adaptive Signal Control- Kővári et al. (2021)

Agent Controller: Kővári et al. (2021) introduced a unique control strategy for a deep Q-learning network (DQN) agent and a reinforcement learning agent with policy gradient (RL-PG). The state input for both agents corresponded to the capacity ingress approach at the intersection level. The reward policy has two (2) factors: (i) normalised standard deviation of occupancy distribution among the incoming lanes of the intersection to the traffic flow, and (ii) punishment factor for prioritising empty flow direction. The agent has an action size of two (2).

Experimental Design: The testing model is an isolated intersection environment. The single-lane junction has four (4) directional flows with two (2) turning movements (through and right). The lane length is 500 metres, and the authors assume these lanes are fully monitored using loop detectors. The minimum phase duration is 30 seconds. The simulation deploys random,

uniform traffic flow with a frequency of 1veh/sec. A time-loss-actuated controller was used to test the proposed control policy.

Results: The results showed that the proposed occupancy policy surpassed the performance of the actuated controller in all measured attributes except travel time. The travel time showed that the actuated controller had a close performance to the RL-PG agent (1%), and it surpassed the DQN agent (by 6%). DQN performs better than the actuated controller in reducing waiting time (20%), and queue length (19%). Though the control policy based on measuring standard deviation can be effective for apparent traffic flow concentration, in low flow scenarios, such a difference might diminish, making the controller stuck in deploying an unfavourable decision. In addition, the controller strategy corresponds to vehicle-related measures and does not address the optimisation of intersection capacity.

2.7.2 Double Dueling Deep Q Network (3DQN)

2.7.2.1 Double Dueling Deep Q-network (3DQN) with Prioritized Experience Reply- Liang et al. (2019)

Agent Controller: Liang et al. (2019) initiated a double dueling deep Q network with prioritised experience reply (3DQN). The integration between double DQN and dueling DQN reduces the possibility of overestimation and improves performance. The state representation is based on DTSE for vehicle position and velocity. The traffic policy aims to minimise the cumulative delay

between two (2) cycles. The action space is size four (4). The reward is a function of the cumulative waiting time between two neighbouring cycles.

Experimental Design: The authors designed a 4-way isolated intersection with three (3) lanes at each of the four (4) directions. Each direction allowed for three (3) turn movements (left, through, and right). The lane length is 150 metres, and the vehicle size is five (5) metres with two (2) metres of minimum gap spacing. The traffic flow per lane was fixed to a 1/10 frequency rate, or one (1) vehicle every 10 seconds. The phase allocation corresponded to a cycle plan, but the phase duration corresponds to a minimum of zero (0) seconds (phase-skip) and a maximum of 60 seconds. The proposed controller extends or terminates the phase duration every five (5) seconds.

For examining the proposed system, the study compared four (4) systems, including two (2) fixed controllers with pre-timed phase durations of 30 and 40 seconds, the adaptive traffic signal control (ATSC) from Pandit et al. (2013), and the deep Q-network (DQN) with auto-encoder by Li et al. (2016).

Results: The authors evaluated the training session's performance. Regarding average waiting time, the 3DQN achieved about 9 seconds of saving compared to fixed controllers. The online ATSC algorithm's performance is limited as it lacks foresight for future demand. The DQN's performance was unstable as it relied on the queue length attribute. The queue

length did not capture traffic conditions accurately, as explained by the authors. Overall, the training evaluation for the average waiting time showed that the proposed 3DQN agent was stable and outperformed other strategies by 20%. The work is limited to training; the 3DQN was not tested for convergence. During the training examination, it was noticed that the proposed agent's variations (3DQN, no double, no duelling, no prioritisation) reached a comparable cumulative reward near the 1,050 episode mark.

2.7.2.2 Traffic Policy Using High-Resolution Event-Based Data for Adaptive System – Wang et al. (2019)

Wang et al. (2019) extended a double dueling deep Q network for an adaptive system using a novel traffic policy (3DQN). Three (3) representative states' input was retrieved from the environment, including phase signal, vehicle position, and occupancy. The action phase was acyclic, where the agent chose from four (4) fixed signal phase plans. The multi-objective reward function was based on factored measurements of throughput and waiting time with trade-off coefficients for vehicles at the intersection. Therefore, the agent aimed to maximise the throughput and minimise the trip delay. The architecture comprised three (3) convolution layers, followed by two (2) fully connected layers, and then one (1) layer for each value and advantage function for the dueling network. The output is one (1) layer for action size.

Experimental Design: The authors observed the state of the environment via three (3) inductive loop detectors. The first detector is placed

at the stop line to record vehicle throughput; the second detector is setback at a distance of 51 metres from the stop line; and the third detector is placed at one (1) metre from the beginning of the lane. Each approach had a similar detector configuration. The study set the green and yellow phases to four (4) seconds each. The vehicle length was uniform at four (4) metres with a minimum headway gap of two (2) metres. The car following model corresponded to the Krauss model and followed the Poisson process. The authors implement various vehicular arrival rates every 15 minutes for the 1.5-hour model. The quarter-hour flow rate ranged from 180 to 360 vehicles per hour.

The testing was carried out on a similar junction configuration to Liang et al. (2017). To evaluate the performance of the proposed 3DQN, a fixed time and a fully actuated controller were added to the study. The fixed time had effective green time ranges between 27 and 30 seconds. The actuated controller executed a minimum phase duration starting from 17 seconds to a maximum duration of 32 seconds.

Results: The evaluation performance indicated that the proposed system exceeds the fixed and actuated controllers in reducing the average vehicle delay by 21.2% and 10.1%, respectively, reducing queue length by 29.7% and 16.4%, respectively, and increasing average vehicle speed by 15.5% and 6.9%, respectively. There is a significant difference in phasing time between the proposed 3DQN and the conventional controllers. The 3DQN had a very short phase duration compared to the comparative controllers. The low

phase duration is the advantage of the 3DQN system, as it ultimately reduces the waiting time and shortens the queue length.

2.7.2.3 Decentralised Network Adaptive Signal Control by Mulit-Agent Deep Reinforcement Learning- Gong et al. (2019)

Control Agent: Gong et al. (2019) proposed a double dueling deep Q-network to optimise the signal problem (3DGN). The decentralised system assigns one agent to each intersection. The control agents are coordinated with each other to share information (traffic data and signal state) and achieve better network optimisation. The state is two (2) vectors: a vehicle's location and a signal phase state. The controller actioned an acyclic programme for managing the intersection's traffic flow. The control policy (reward) aims to minimise the cumulative waiting time of vehicles in the queue.

Experimental Design: The authors experimented with an arterial network comprising eight (8) intersections. The intersections were modelled after an actual traffic network in Seminole County, Florida, US. The two-hour model had a total traffic flow of nearly 13,200 vehicles. The calibration indicated that the root mean square error (RMSE) is 4.7 vehicles per hour between the simulated and real counts. The training and testing were conducted in the same suburban simulation set but with a random state for the fixed traffic flow. The data input was monitored within 90 metres of lane length.

Results: The proposed decentralised 3DQN algorithm improved travel time experience by 10.27% and reduced travel delay by 46.46% compared to the actuated controller. On the other hand, the proposed system increased the number of stops by 11.29%. The authors justified this increment as the 3DQN controller's objective is to mitigate the delay. Such a control policy gave a fair travelling right between major and minor approaches and through and turning movements. The fair right of the movement led to force-stopping vehicles at major approaches. The control policy is limited to serving narrow aspects of traffic conditions and trading on other traffic conditions.

2.7.3 Deep Stacked Autoencoders (SAE) Neural Network- Li et al. (2016)

Agent: Li et al. (2016) introduced a deep stacked autoencoders (SAE) neural network for Q-learning algorithm. The SAE consists of two (2) hidden layers. An autoencoder sets the target output layer equal to the input layer with the sigmoid activation function. The state of the environment is represented by queue lengths. The reward function corresponds to the absolute value difference in queue length between the competing directions.

Experimental Design: The authors tested the proposed SAE system on a cross intersection with a simple through movement only (no U-turn, right, and left turns are allowed). Furthermore, the signal operation is simple, with two (2) phases for opposing direction movement and no red phase clearance. Each approach had a random traffic flow volume between 100veh/hr and

2,000veh/hr. The green phase corresponds to a minimum of 15 seconds, and extension is allowed. The authors compared their proposed deep Q-learning based on the SAE system and a conventional Q-learning.

Results: The findings showed that the SAE system outperformed the conventional system by reducing the mean delay time by 14%, and mean values for the number of stopped vehicles were reduced by 410 during the entire simulation run. Regarding the queue lengths, both systems had a similar effect on producing balanced queues. The limitation of this work is related to simplifying intersection movement to one (1) direction only. The efficiency of the SAE in achieving the control policy (queue length) is also limited.

2.7.4 Deep Deterministic Policy Gradient (DDPG) Reinforcement Learning - Casas (2017)

Agent: Casas (2017) developed a deep deterministic policy gradient (DDPG) for a continuous state-action urban traffic light control environment. Vehicle information, including vehicle counts, average speed, and occupancy percentage, is used for state representation. The author proposed a unique approach to operating the traffic light by controlling the phasing duration but fixing the cycle time and phase order. This time plan was programmed by capping the phasing duration to 80% of the cycle time using a phase adjustment matrix. The reward function is based on a score factor for speed scaled by counts of vehicles and a discount factor to restrain rewards in the range [-1,+1]. The activation function is rectified linear activation (ReLU).

Experimental Design: The author considered three (3) testing scenarios to evaluate the proposed DDPG against conventional multi-agent Q-learning and a random signal phasing generated by the micro-simulation software. The one (1) hour testing simulation environments are: (i) a 4-way isolated intersection with through and right turn movements only and traffic flow of 150veh/hr, (ii) a 3x2 hypothesised grid network with six (6) intersections and three movement permissions, i.e., left, right, and through, with a random traffic flow, and (iii) a network representing the Sants area in Barcelona, Spain, with 43 junctions and traffic demand matching the peak hour in the study area. The state input was obtained using loop detectors.

Results: The author refers to the performance of DDPG as superior to classical Q-learning in scenario 1 (simplest case), slightly better in scenario 2, and at the same level for scenario 3 (real-world network). The convergence of DDPG is linearly proportional to the scenario level of complexity, and a longer training time could have led to better performance of the proposed technique. The author stated that DDPG addressed the curse of dimensionality better than classical Q-learning. It is unclear if the chosen reward scheme was fair and whether the placement of road detectors assisted in the performance of DDPG. The arrangement of detectors in the first two (2) situations is similar. In contrast, the placement of detectors was random in the real-traffic model of the network.

2.7.5 Multi-agent Deep Q-learning Agent (MADQN) - Rasheed et al. (2020)

Control Agent: Rasheed et al. (2020) proposed a multi-agent deep Q-learning agent (MADQN) to optimise signalised intersections under disturbed traffic flow. Four (4) inputs represent the state: queue length of all incoming lanes at the controlled intersection, queue lengths of all incoming lanes at the neighbouring intersection, elapsed red timing at the stopped approach of the controlled intersection, and rainfall intensity (disturbance level). The latter was scaled into five (5) levels, ranging from no rain to a maximum value of heavy rain. The action was chosen from pre-determined traffic phases. It is unknown if the controller adjusts the timing. The reward function is the difference between the waiting time between past and present actions.

Experimental Design: The authors simulated two network models for the environment: (i) a real network and (ii) a grid network. The real-traffic network model comprised seven (7) intersections in Sunway City, Selangor, Malaysia. The grid network consisted of nine (9) intersections. The Bur type XII function was utilised for vehicle arrival. This function is suitable for traffic disturbances, particularly rainfall. No information about the hourly traffic flow was provided in the study.

For comparison, the authors used three (3) signal controllers, including deterministic, RL, and MARL. Two (2) traffic scenarios, including (i) a recurring congestion and (ii) a non-recurring congestion, were modelled. The

recurring congestion scenario resulted from increased traffic volume during peak hours.

Results: The results showed that the proposed MADQN converged faster than the MARL during training for both scenarios. The testing indicates that the proposed agent surpassed other techniques by increasing throughputs (70%), reducing queue lengths (75%), and lowering waiting times (70%).

2.7.6 End-to-End Policy for Deep Learning Controllers

2.7.6.1 Deep Dueling On-policy SARSA Learning Agent-Yen et al. (2020)

Control Agent: Yen et al. (2020) were the first to apply a deep dueling agent for an on-policy SARSA approach to coordinate a network of intersections (2DSARSA). The state represented traffic flow maps (TFM); traffic flow in terms of delays. The TFM is an image encoding the Head-of-Line (HOL) of sojourn times of each lane at each intersection. The HOL sojourn times differ between adjacent intersections. The reward function is a power metric defined by the throughput ratio to the average end-to-end delay. The action function reflected a phase activation function.

Experimental Design: The authors test the system in a 3x3 grid matrix (9 intersections) with a simple action plan size of two (2). For testing, the authors developed two (2) architectures (single and dueling) for SARSA and Q-learning. The singles are deep learning agents, including DQN and

DSARSA. The dueling structures are 3DQN and the proposed 2DSARSA. In addition, to evaluate the performance, two (2) back pressure controllers (BPC) were used with different control functions: delay (DBPC) and queue (QBPC).

Results: The results indicated that the proposed 2DSARSA and DSARSA learn effectively from the environment and have faster convergence and stability during the learning process. This quick convergence was expected as SARSA is an on-line policy that relies on the initial policy. In comparison, the Q-learning algorithms (3DQN and DQN) are off-line policies, and they need to learn the actions from the policy. This reported finding is not supervising due to the different structure of both algorithm categories. Also, the comparison was based on training convergence, but the ability of the proposed 2DSARSA to perform in a test environment was not convincingly proven.

The queue-based back pressure (QBPC) showed better performance in end-to-end delay based on box plot analyses. The QBPC showed a better inter-quartile median, minimum, and maximum) range (compared to the proposed 2DSARSA and delay-based back pressure (DBPC). The 2DSARSA showed smaller outliers in box-plot analyses. The training showed that the 2DSARSA algorithm had reached an average end-to-end delay of as high as 350 seconds. Overall, the centralised system (2DSARSA) versus the decentralised system (QBPC) did not show outperformance in end-to-end delay, even though the 2DSARSA utilised delay to mitigate intersection operation.

2.7.6.2 Deep RL-Background Removal ResNet (BGR ResNet)-Chu et al. (2021)

Agent: Chu et al. (2021) presented a unique end-to-end off-policy deep reinforcement learning. The system aims to develop a low-cost system based on images captured by surveillance cameras. A background removal ResNet (BGR ResNet) is added to the system to reduce its complexity. These devices are assumed to partially observe the road state condition, such as vehicle position, speed, and queue length, and transmit them to the agent controller. The control strategy is to minimise the average waiting time at the intersection. The agent determines the optimal action from two (2) possible phase configurations.

Experimental Design: The authors adopted a real-world intersection and synthetic scenarios for testing. The isolated intersection is a four-way junction in Cologne, Italy. The site data indicated a total traffic flow of 1,800 vehicles per hour for the modelled intersection. In addition, four (4) generated scenarios based on random traffic flows were tested, including 2,000 vehicles per hour (S1), 1,500 vehicles per hour (S2), 1,250 vehicles per hour (S3), and 1,150 vehicles per hour (S4). The S1 had equal traffic flow for all directions (500 vehicles per hour per direction). The other scenarios differentiated flows among directions into major and minor flows.

The proposed BGR ResNet system comparison included: a fixed signal, a max pressure, a greedy algorithm based on fleet size, a greedy

algorithm based on waiting time, a DQN logic (Mnih et al., 2015), a C51 (Bellemare et al., 2017), and a Rainbow (Hessel et al., 2018). Two (2) architectures are developed for each RL agent, including convolution neural network (CNN) and ResNet.

Results: Overall, the proposed BGR ResNet system surpassed others in all but one (1) scenario. In this scenario (S4), the greedy algorithm based on waiting time reacted to the approaches with higher traffic to relieve the pressure that eventually led to surpassing the performance of the BGR ResNet system. The average waiting time delay for all scenarios is improved by at least 13% compared to the closest rival (greedy algorithm).

The comparison between the agent's structure for CNN and ResNet did not show regularity. For instance, DQN-CNN surpassed DQN-ResNet in Cologne, S1, and S2 scenarios, and it showed an improvement of 4.8% in average waiting time for all scenarios. A similar pattern was found in C51, where the CNN design outperformed ResNet in all scenarios but S3 and S4 scenarios and achieved a 12.2% lower average delay. In contrast, the Rainbow-ResNet performed better than the Rainbow-CNN in all scenarios and achieved 29% savings in the timing attribute for all five (5) tested scenarios. From these reported measures, the contribution of the agent's architecture to the agent's performance was not evident.

Moreover, based on the reported results, it is noticed that the performance measure in terms of average delay across different systems

closed up with low traffic flow. The authors ran 10 minutes of simulation scenarios; this training time is probably insufficient to converge the proposed systems to their optimal potential and caused poorer performance at an imbalanced traffic ratio. The impact of penetration rate is not explored in the study. The camera angle is assumed to cover all roads at the intersection. Besides that, the authors assumed that vehicles transmitted their waiting time to the controller (connected vehicle technology).

2.8 Review of Current Challenges for Deep Reinforcement Learning Controllers

The DRL signal controllers gained momentum after their initial introduction by Genders and Razavi (2016). A present review of the DRL studies indicated that many advanced methods were utilised. Six (6) architectures were identified, including DQL, 3DQN, SAE, DDPG, MADQN, and DSARSA. Several findings require attention in terms of agent architecture, environment model, and control strategy.

2.8.1 Agent Architecture

In terms of agent architecture, the review of the DL controller studies does not conclusively prove the superiority of a particular **DL type**. Limited studies have incorporated the testing and comparison of various DL structures. For instance, the earliest study by Genders and Razavi (2016) compared two (2) neural network architectures. The first neural agent was based on three (3)

hidden layers of convolution with experience reply (DQTSCA), and the second agent was one (1) shallow hidden layer without experience reply (STSCA). In terms of results, the authors reported that both systems had an insignificant difference in throughput, and the outstanding performance of DQTSCA in mitigating delay was related to its own reward function (cumulative delay).

Another issue associated with control agent design is **sustenance** in control decisions. The agent converges to a wrong decision under certain circumstances. Casas (2017) indicated that the deep learning agent converged to conventional Q-learning, Li et al. (2016) stated that the DL system had limited significance to enhance attributes of traffic performance, and Chu et al. (2021) reported underperformance of the agent in imbalanced conditions of traffic flow.

The current review shows that simple DQN achieved better performance than complex DQN, particularly 3DQN, MADQN, 2DSARSA, and ReNEST. Figure 2.7 compares the cited works of literature regarding waiting time and travel time improvements.

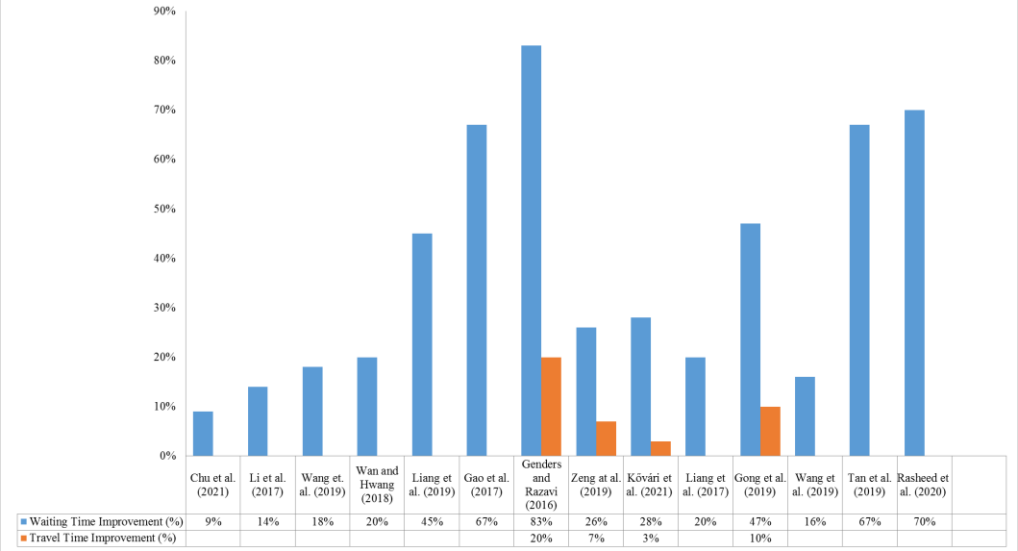


Figure 2.7: DRL controller studies and performance

2.8.2 Environment Model

The DL agent requires training and then testing. Given that the RL agent learns from the model context, it is a requisite that the model of environment imitates the actual conditions and estimate precise changes in correspondence to the decisions made by the agent (Han, 2018). Both of these tasks are achieved using a suitable concept of environment. The review of current studies pointed out the following major **evaluation** issues:

1. Under-represented traffic model. Most of the current developments in ML techniques focus on hypothesised isolated intersections. The 4-way junction is further simplified to controlled approaches only and non-conflicting movements. The stochasticity of the traffic environment is not accurately addressed. The studies anticipate

similar drivers' characteristics and do not test mixed-mode travel corridors.

2. The simplicity of intersection movement. Many authors have mainly focused on isolated intersection and grid network models. Both of these environments simplify the complexity of the traffic environment and the control mission. Few studies tested arterial networks. Unfortunately, their systems were unsuitable for network operation, despite passing the isolated junction, as in Casas (2017).

3. Split optimisation. The control plan is usually associated with phase allocation. The signal controller re-locates, terminates, or skips a phase in a cyclic arrangement. A few studies considered phase duration. Split durations are very small (<5 seconds). Such duration does not adhere to safe operation in the real world and is incompatible with driver expectations.

4. Traffic flow. Most researchers develop a network model with a clear distinction between traffic hierarchies (major and minor). Using primary and minor flows is not always representative of a real-world scenario. The traffic flow hierarchy diminishes at major urban intersections, and vehicles compete for an equal right of passage in actual driving conditions. Another pressing model issue is that traffic flows are presumed to be constant. Such a design concept also contradicts natural flow characteristics. The flow

saturation is low (<3,200veh/hr). The investigation of various demands is rare.

The experimental design setting gives the agent an advantage. This is because training and testing are carried out on the same platform for the memory-based controller. By default, the DRL agent passes the test scenario. Nevertheless, this does not mean that the DRL is being validated accurately. When testing condition changes, the DRL tends to show unstable performance as in Casas (2017), Chu et al. (2019), Wan and Hwang (2018). Therefore, **validating** the ability of DL applications in stochastic traffic environments is questioned and requires study.

2.8.3 Control and Reward

The review of RL and DL agents indicates that researchers do not segregate between state input and reward feedback. Both of these parameters are similarly defined using environmental dynamics. As it is challenging to deal with dynamic attributes, the utilised reward strategies eventually lead to complex agent design and unworkable assumptions for current infrastructure readiness.

For network optimisation, researchers presented DL systems requiring global coordination and centralised systems for decision-making. The multiplex design ensures that traffic states and rewards are monitored closely. Researchers assumed infinite space and storage capacity to accommodate

vehicles in isolated intersection situations. There is no specified boundary for the DL controller to mitigate signal operation. No research has yet proposed an efficient control strategy to optimise the junction's capacity directly.

The following **Table 2.6** provides a summary of research work in the DRL and its application in adaptive traffic signal controllers.

Table 2.6: Summary of literature review for deep reinforcement learning (DRL)agents

No.	Author	Agent	State	Action	Traffic Control Strategy (Reward)	Simulation Settings				Comparative System(s)	System Control	Communication Channel	Improvement	Limitation
						Test Bed	Traffic Flow Distribution	Traffic Flow Volume (veh/hr)	Phase Plan					
1	Genders and Razavi (2016)	DQTSCA	DTSE	Phase allocation	Cumulative delay of all vehicles at intersection	Isolated intersection (hypothetical)	Inverse Weibull: left and right turns, Burr: through traffic	Inverse Weibull: 0-150 Burr:250-450 Total (max): 3,000	4 Phases = {EW,NS, NSL, EWL}	TSCA	Single	NA	83% lower cumulative delay 66%shorter queue length, 20% lower travel time	Control policy, different state representation, constant traffic condition
2	Gao et al. (2017)	DQN	DTSE	Phase allocation	Cumulative waiting time of all vehicles at intersection	Isolated intersection (hypothetical)	Bernoulli process	Left turn = 1/10, through major = 1/5, through minor = 1/10, Total (max): 2,280	2 Phases = {EW,NS}	LQFA & Fixed controllers	Single	NA	47%-86% lower delay time	Stability examination, constant traffic condition
3	Zeng et al. (2019)	MQN	DTSE	Phase extension, phase skip	(i) number of vehicles passing stop line, (ii) number of halting vehicles at green phase direction, (iii) phase skip punishment, (iv) normalised halting vehicles at intersection level, and (v) total waiting time of vehicles around the intersection.	Isolated intersection (hypothetical)	Fixed*	Mean configuration I#: major: 1,800, minor: 1,200, total: 3,000 mean configuration II#: 4,800	4 Phases = {EW,NS, NSL, EWL}	Fixed & DQN	Single	NA	Compared to Fixed, the MQN had 7% lower average delay, 27% shorter queue length, 7% average travel time, 26% lower average waiting time.	Superiority of MQN is not evident versus DQN
4	Kővári et al. (2021)	DQN & RL-PG	Occupancy	Phase allocation	(i) normalised standard deviation of occupancy distribution among the incoming lanes of the intersection to the traffic flow, and (ii) punishment factor for prioritising empty flow direction	Isolated intersection (hypothetical)	Random	300	2 Phases = {EW,NS}	Time loss actuated Controller	Single	Loop detectors	3% lower travel time , 28% lower waiting time, 18% lower queue length, 8% lower emissions, 7% lower fuel consumption	control policy requires conceivable traffic flow difference to function effectively
5	Liang et al. (2017)	3DQN	DTSE={Vehicle's position and speed}	Phase duration	cumulative waiting time between two neighbouring cycles	Isolated Intersection (hypothetical)	Random	Major: 1,440 Minor: 720 Total:2,160	4 Phases = {E,W,N, S}	fixed controllers, ATSC, DQN auto-encoder	Single	NA	20% lower waiting time in training	No testing was carried out to verify agent's performance stability
6	Gong et al. (2019)	3DQN	DTSE= {Vehicle's position and phase status}	Phase extension, phase skip	cumulative waiting time of vehicles in queue	8 intersections (modelled from real scenario)	Calibrated using real traffic condition	13,175	3 & 4 phases	actuated controller	Mutli-agent	Connected vehicle	10.27% lower trave time, 46.46% reduced total delay	Control strategy trade-off

No.	Author	Agent	State	Action	Traffic Control Strategy (Reward)	Simulation Settings				Comparative System(s)	System Control	Communication Channel	Improvement	Limitation
						Test Bed	Traffic Flow Distribution	Traffic Flow Volume (veh/hr)	Phase Plan					
7	Wang et al. (2019)	3DQN	DTSE={Vehicle's position and occupancy}	Phase allocation	maximise the throughput and minimise the trip delay	Isolated intersection (hypothetical)	Poisson process	3,250	4 Phases = {EW,NS, NSL, EWL}	Fixed and actuated controller	Single	Inductive loops	lower waiting time (21.2%-fixed, 10.1% actuated) reduced queue length (29.7% fixed, 16.4% actuated) higher vehicle speed (15.5% fixed, 6.9% actuated)	Phase timing is shorter for the 3DQN versus comparative systems
8	Tan et al. (2019)	Coder	Queue length	Phase allocation	balancing queue length and moving vehicles	Grid network B: 4x3 Grid network C: 4x6 [hypothetical]	Evenly distributed	Grid B: 904, grid C:1,344	2 Phases = {EW,NS}	fixed, random, linear Q-learning, and R-DRL	Mutli-agent	NA	Network B: lower average queue length by 33%, and shorted waiting time by average of 71%. Network C, lower average queue length by 28%, and shorter delay by 63%.	Sub-region size, & MDP environment
9	Wan and Hwang (2018)	DQN with discount factor	DTSE= {Vehicle's position and phase status}	Phase allocation and duration	Cumulative time delay between 2 actions	Isolated intersection (hypothetical)	Random	Undersaturated: 800/direction, Over-saturated: major 2,000/direction and minor: 100/direction	8 phases = {N, S, E, W, NL, SL, WL, EL}##	DQN with 1 input and Fixed	Single	NA	Undersaturated: 20% lower delay, (not reported) Oversatruated:15% lower delay, 17% higher throughput	MDP environment & experiment settings
10	Casas (2017)	DDPG RL	Vehicle counts, speed and occupancy	Phase duration	Factor for speed score scaled by a discount factor and vehicle counts	Isolated intersection (hypothetical), 3x2 hypothesised grid network (hypothetical) and 43 junctions' network (Real network)	Random	Isolated intersection: 300, hypothesised and real network: NA	Isolated intersection (no Left turn), 2 phases = {EW,NS}, hypothesised grid network varies 4-6 phases, 43 junctions network varies 1-6 phases mainly 2 phases	Random algorithm and multi-agent Q-learning	Single	Loop detectors	DDPG as superior Q-learning in scenario 1 (simplest case), slightly better in scenario 2, & same level for scenario 3 (real world network). Both RL surpassed random algorithm	Experimental settings to verify superiority of DDPG vs. RL
11	Li et al. (2017)	SAE	Queue length	Phase duration	difference of queue length between competing directions	Isolated intersection (hypothetical)	Random	100 to 2,000 per approach (only through is allowed)	2 Phases = {EW,NS}	Q-Learning	Single	NA	SAE system: 14% lower delay time, 410 lowered stops.	2DSARSA lacks to mitigate operation based on its reward policy compared to QBPC

No.	Author	Agent	State	Action	Traffic Control Strategy (Reward)	Simulation Settings				Comparative System(s)	System Control	Communication Channel	Improvement	Limitation
						Test Bed	Traffic Flow Distribution	Traffic Flow Volume (veh/hr)	Phase Plan					
12	Yen et al. (2020)	2DSARSA	traffic flow in terms of delays	Phase allocation	power ratio (throughput: end-to-end delay)	3x3 grid matrix (9 intersections) (hypothetical)	Poisson process	minor (2 directions): 135, moderate 338, and major 675. Total=1,283	2 Phases = {EW,NS}	Deep Q-Learning (2DQN, DQN), Deep SARSA, back pressure (DBPC, QBPC)	Multi-agent	Connected vehicle	QBPC: ~20% lower median end-to-end delay vs. 2DSARSA	Experimental settings to verify superiority of 2DSARSA vs. BPC, extensive environment knowledge
13	Rasheed et al. (2020)	MADQN	Queue length, rainfall intensity, and red time	Phase allocation	difference in waiting time between two (2) actions	7 signalised intersection (real) 3x3 grid network (hypothetical)	NA	NA	Varies: 2-4 phases	Deteministic, RL, MARL	Multi-agent	NA	MADQN: 70% lower throughput, 75% lower queue length, 70% lower waiting time	Experimental settings, extensive environment knowledge
14	Chu et al. (2021)	DQN-BGR ResNet	Traffic image	Phase allocation	Minimise the average waiting time	Isolated intersection with real and hypothetical traffic flow	NA	2,286	2 Phases = {EW,NS}	Fixed signal, Max pressure, Greedy-based fleet size, Greedy-based waiting time, DQN, C51, and Rainbow	Single	Surveillance cameras and connected vehicles	DQN-BGR ResNet: >13% lower waiting time	Experimental settings, extensive environment knowledge

EW: Green for East and West approaches, NS: Green for North and South approaches, EWL: Green for East and West approaches with Green for Left turning movements, NSL: Green for North and South approaches with Green for Left turning movements, E: Exclusive green for East approach only, W: Exclusive green for West approach only, N: Exclusive green for North approach only, S: Exclusive green for South approach only

*assumed fixed arrival rate as not mentioned in the study

#configuration II is equally distributed among flow all directions

phase length as short as 1 second, N, S, E, W = Exclusive through and/or right turning, NL, SL, EL, WL = Exclusive left turning only

2.9 Summary

2.9.1 Adaptive Control Systems

The review of adaptive controllers revealed attractive methods to deal with a traffic light. Many of these approaches are restricted to the conceptual level and suffer from drawbacks that restrict them to prototypes. Therefore, real-world scenarios pose a challenge due to their complexity. There are three (3) areas of improvement that require primary intervention in the field of adaptive controllers:

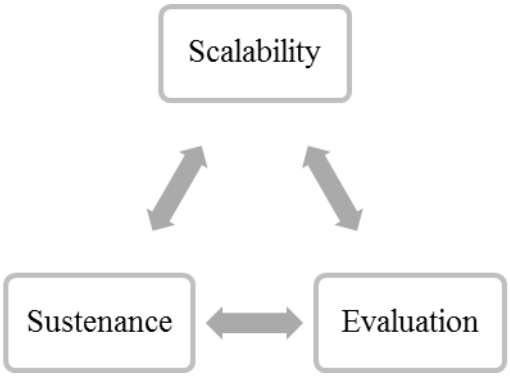


Figure 2.8: Adaptive system improvement cycle

1**. Scalability** of the adaptive system in realistic environment application. Two (2) aspects of improvements are needed.

Aspect 1: The simplicity of system design to mitigate network operations. The solo agent system performs independently and is less complex than the multi-agent system. The single agent is slow in converging to an optimal decision in online learning. Vibrant traffic dynamics could bring the

operation to its knees. In this regard, system developers resorted to ML and AI (offline) techniques to equip the single agent with a prior understanding of the environment assignment using training. Based on studies, memory-based control systems surpassed online agents in mitigating signal controllers.

Aspect 2: Though much development in computing power has been achieved recently, the communication-based approach is not visible, at least for the current time. Connected and autonomous vehicle technology is far from being reached in the coming years. In addition, the function approximation for traffic stare is a coarse representation that ignores the stochasticity and heterogeneity of traffic nature. Hence, real-time communication is recommended to deal with the signal problem using intelligent adaptive controllers. Traffic congestion is an issue, and using available technology is necessary to make the signal controller more practical.

2. **Sustenance** of the system to achieve complete adaptation without human interface. Two (2) aspects require intervention and implementation.

Aspect 1: Agents are the traffic signals, but the learning task is formulated for feature extraction. The system utility is measured from the perspective of vehicle objectives, and this presents a core challenge for the signal controller. The signal controller has to deal with ever-changing dynamics. Such limitations restrict the implementation of adaptive controllers in an actual traffic situation. Instead, the control strategy must mitigate flow rates based on available capacity at travel corridors.

Aspect 2: The adaptation assignment should control the cycle operation. Restricting the role of the adaptive controller to phase duration or sequencing only narrows the applicability of the controller to adapt to surrounding dynamics. Thus, the acyclic technique is recommended for adaptive controllers.

3**. Evaluation** of the system in a representative model context. Including a high level of realistic details in the simulated models is necessary (Gao et al., 2016). The proposed systems' credibility remains questionable without validating the simulation (Bazzan and Klügl, 2014). To validate a system, thorough testing is required to determine how well the controller system corresponds to the natural environment. In addition, there is a need to develop flexible and robust systems capable of treating complex simulation scenarios.

2.9.2 Deep Reinforcement Learning Controllers

The review of DRL studies showed a promising controller type that could revolutionise adaptive system generation. The researchers' experiments strongly suggested that the DRL outperformed other generations of traffic signal systems and online adaptive controllers. On the other hand, the current study direction in DRL agents suffers from similar major issues in adaptive control studies, including **scalability**, **sustenance**, and proper **evaluation**.

Given the growing complexity of DRL techniques, new techniques must be deployed, and the mechanisms of the traffic environment must be better understood for the whole system to become more efficient. Innovative control strategies are needed to address intersection capacity rather than the standard feature extraction approach. The latter technique has reduced the role of DRL to responsive rather than adaptive systems. Integrating a junction-based strategy is anticipated to be more effective. Congestion at an intersection is a result of insufficient capacity (Tiaprasert et al., 2015).

Given that the DRL agent learns using a trial-and-error process, it is of prominent significance that the modelled environment reflects the actual traffic conditions and estimates accurate dynamics in response to the agent's decision (Han, 2018). Nonetheless, most presented modelling approaches do not consider aspects of the actual traffic environment. It is imperative to perform research on signal control theory based on the multimodal traffic environment (Wang et al., 2018). The current benchmarking approach is hard to prove for real-world applications (Gong et al., 2019).

CHAPTER 3

SYSTEM FEATURES AND CONTROL POLICY

This chapter introduces the features of the proposed traffic signal control in this study. Then, a novel control strategy is presented for the DRL controller. The proposed system design is to tackle the limitations cited in the literature review in Chapter 2.

3.1 System Design Features

Many promising ideas were presented in the present works. However, most of the described methods are still at the conceptual prototype level, and their application in the real-world poses a challenge. In this study, we intend to develop a controller that closes the literature review gaps. Issues related to **scalability**, **sustenance**, and **valuations** are encountered in the DRL systems. Therefore, the design framework must feature (i) a low-cost communication protocol, (ii) an ability to function with minimal knowledge, (iii) an adaptation to traffic dynamics, and (iv) an optimisation for various traffic metrics.

In terms of system design, a single-agent framework is adapted. This is because the solo system is less complex, sufficient to learn optimal decisions, and prevents chaotic behaviour in large-scale implementation. In comparison, coordination and centralised systems incur high set-up costs. The single agent herein refers to a system where each junction is controlled by one (1) agent

only, and this control agent (i) does not communicate with neighbouring intersections and (ii) deploys logic decisions independently. Figure 3.1 presents the recommended system characteristics.

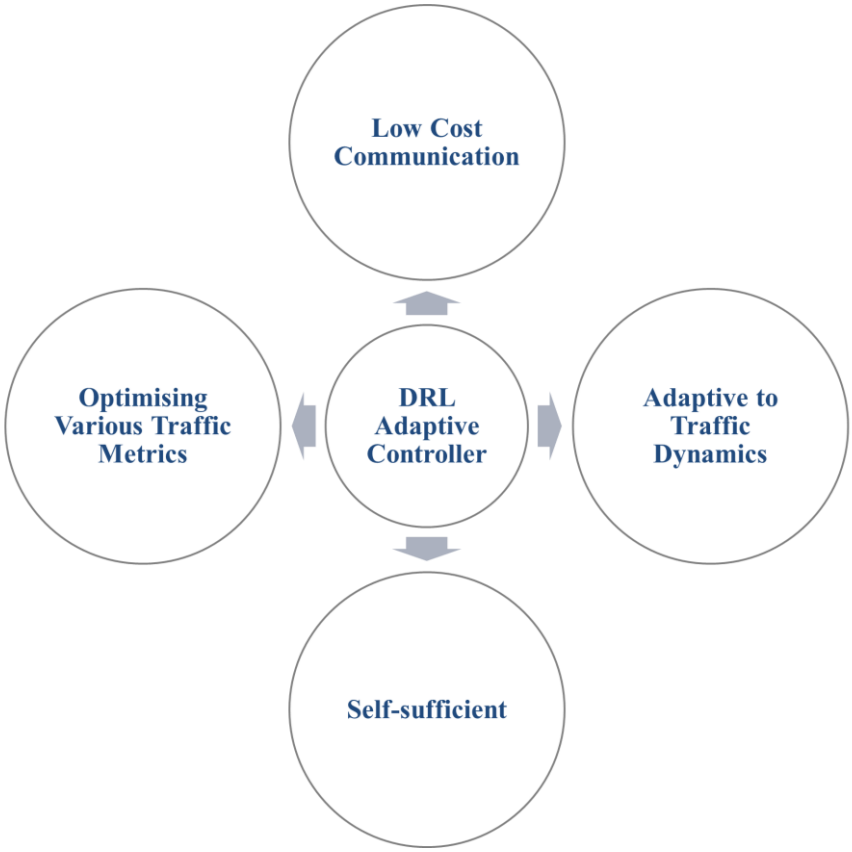


Figure 3.1: Characteristics of DRL controller

The DRL is capable of feature mapping and self-learning and requires a certain model set-up. A model-free RL is an option for system development. The model-free, unlike the model-based, does not require a transition function and acquires knowledge of the unknown using exploration (Mannion et al., 2015). The transition function adds unnecessary complexity and is difficult to determine in highly stochastic problems (El-Tantawy et al., 2013).

Two (2) popular model-free methods in adaptive control systems are Q-learning and State-Action-Reward-State-Action (SARSA). The SARSA (on-policy) follows an initial policy and cannot explore other policies (Yen et al., 2020). In comparison, Q-learning (off-policy) estimates rewards by selecting actions to maximise the expected cumulative reward (Yen et al., 2020). As Q-learning offers exploration and updates policy accordingly, it is often used for developing the signal controller. Q-learning is proven to converge to optimum action values as long as all actions are repeatedly sampled and are represented discretely (Mannion et al., 2015).

3.2 Traffic Control Policy

The existing policies used in DRL controllers seem to lack innovation and capacity to support infrastructure integrity. The recent DRL studies in Section 2.7 focused on responding to vehicle-based features. The lack of an appropriate control strategy made intelligent controllers responsive to vehicle dynamics, requiring intensive data input and complicating design by coordinating or centralising decision-making.

We propose a novel traffic strategy to support the DRL system in mitigating signal control based on intersection-based feedback. Laval et al. (2007) stated that a system will be at equilibrium if the system input accumulation across time is as close to optimal as possible. This understanding forms the basis for this research study's proposed traffic control policy. To achieve the optimal operation level, downstream (outbound) routes must be maintained at an acceptable level of service. Congestion occurs when the road

section exceeds its desirable density. Considering downstream-based policies are presented in other traffic engineering fields, particularly tunnel (Gazis, 1972) and ramp metering assignments (Khoo, 2011), this is the first time such a strategy is proposed to solve the DRL signal problem.

There are two (2) challenges with junction layout. First, the urban signal intersection represents a multi-directional traffic flow challenge as opposed to the tunnel and off-ramp metering (1-directional traffic flow). Second, the junction configuration often includes unsignalled slip lanes. Figure 3.2 presents each direction of travel as a tunnel scheduling challenge.

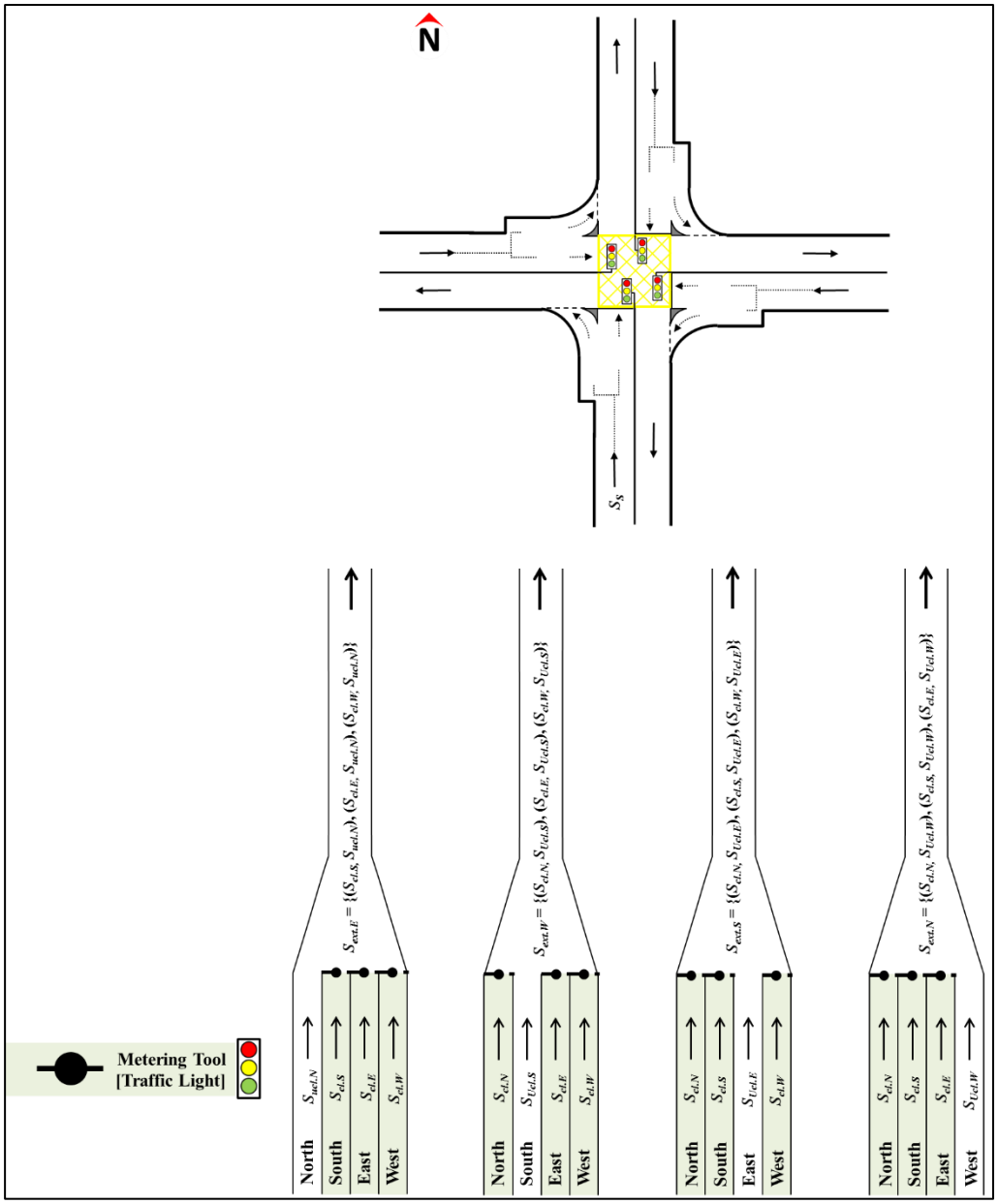


Figure 3.2: Traffic streams at signal junction viewed as tunnelling movements (Left-Hand Traffic)

Each direction has four (4) allowed turning movements (left turn, right turn, through, and U-turn). The left turn guides an uncontrolled traffic stream S_{ucl} with a short slip lane. The remainder of traffic flows comprises individually metered movement S_{cl} by the traffic controller. The S_{cl} is governed by phasing time and allocation. The signal logic deploys an

executive phase p_{exe} for each direction of travel. Such an intersection layout is typical in the real world.

At time step *t*, a platoon of vehicles S_{cl} passing from the upstream stop line to the downstream destination via the intersection will pass through three (3) zones, as in Figure 3.3. These proposed zones include a junction's zone (Z1), a merging zone (Z2), and a post-merging or recovery zone (Z3). At the edges of these zones, there are four (4) observed mean speed *v* values for the passing vehicular traffic. These *v* measurements can be computed from widely available surveillance tools such as cameras or built-in road detectors within a pre-defined time step *t*. These measured values include (i) entry speed v_{ent} to Z1, (ii) exit speed v_{ext} from Z1 (equivalent to entry speed to Z2), (iii) merge speed v_{merg} at the end of Z2 (or at the entry of Z3), and (iv) depart speed v_{depart} at the edge of Z3. The Z3 is assumed to have free flow condition, and speed limit v_{lmt} can be achieved. The detection area length *l* of these zones is equal. The *l* should not be less than (i) the minimum merging distance and (ii) the minimum allowable distance to reach v_{lmt} for a vehicle accelerating from rest at l=0.

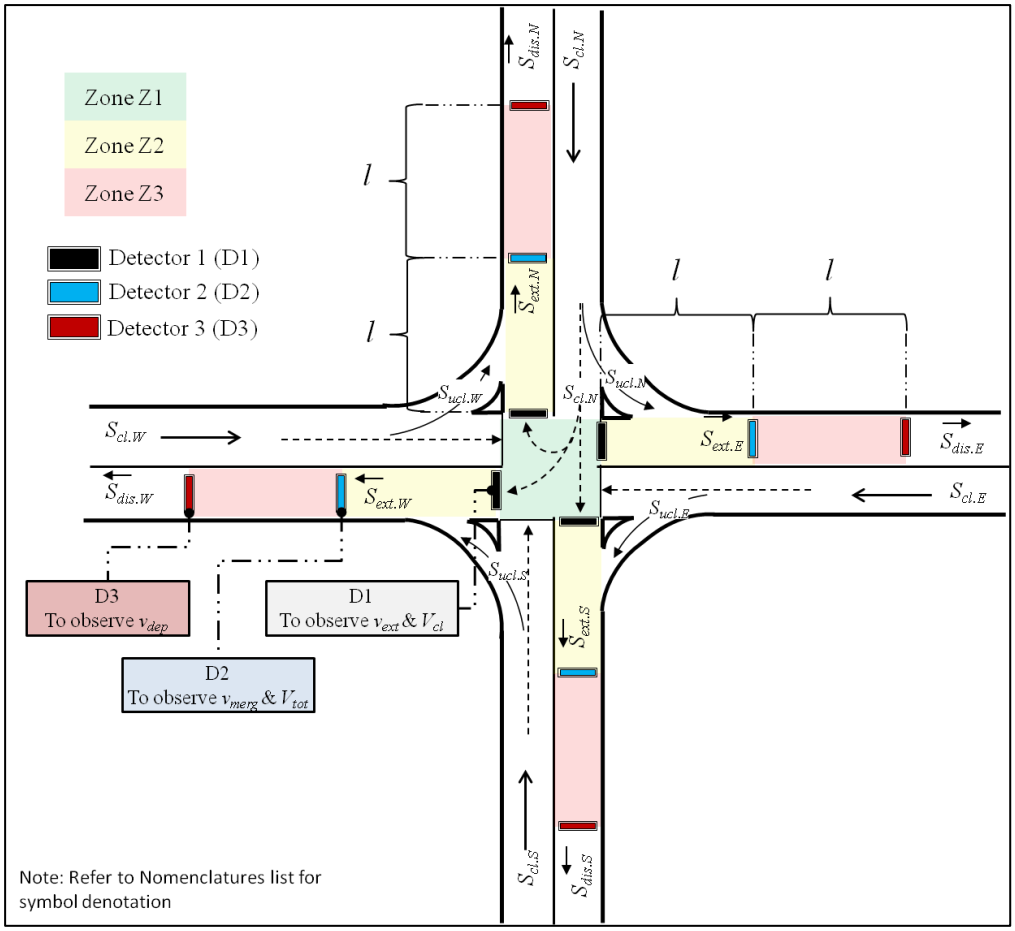


Figure 3.3: Traffic flow entering the intersection from Northbound (Left-Hand Traffic)

The significance of maintaining density at Z3 k_{Z3} to be as close as to its optimal density k_{op} is necessary to ensure that the transition from upstream to downstream is kept balanced at all times ($k_{Z3} \approx k_{op.Z3}$). There are two (2) propositions made for k_{Z3} including (i) $k_{Z3} >> k_{op.Z3}$ and (ii) $k_{Z3} << k_{op.Z3}$. These cases are directly proportional to the arrived flow rate S_{ext}. If case (i) is observed, this suggests that the downstream is overutilized, and the solution is to mitigate the S_{ext} from Z2. On the other hand, if case (ii) is detected, this indicates that the concentration k is insufficient and road capacity at Z3 $C_{sec.Z3}$ is underutilised and is less than optimal. Then, the control system should execute green duration D_{green} to permit cumulative optimum flow passage S_{cl}

only to exit routes. Hence, the formulated strategy questions are (i) how to meter the signal duration D_{eff} to induce the right amount of traffic volume V to ensure optimal density at downstream $k_{opt.dwns}$ is not exceeded (Obj. 1), and (ii) how to assign the correct phase logic p_{exe} among competing arms of signal intersection to deliver throughput V to downstream (Obj. 2).

A set of rules forms the basis of the signal strategy. The fixation of phase allocation at any competing arms of the signalised intersection should fulfil the following two (2) rules.

Rule 1: **Maximising the optimum density at downstream** i.e., $\boldsymbol{max} \sum_{\boldsymbol{n=1}}^{\boldsymbol{N}} \boldsymbol{k}_{\boldsymbol{Z3}}^{\boldsymbol{n}}$ where N is the number of exit destinations per a direction of travel.

The density in Z3 k_{z3} is computed from the following equation 3.1.

$$k_{z3} = \frac{V_{tot}}{l} \tag{3.1}$$

Where V_{tot} is the total number of controlled V_{cl} and uncontrolled V_{ucl} vehicles entering the Z3 region. These measurements can be obtained from the D2 and D3 observer points, as in earlier Figure 3.3.

As the V_{cl} corresponds to effective green time G_{eff}, the maximum allowable throughput $V_{max.cl}$ is computed using the weighted density of

different lanes at downstream $k_{w.dwn}$ within l distance. The following equation 3.2 presents this relationship.

$$V_{max.cl} \leq \frac{k_{w.dwn}}{l} \tag{3.2}$$

Equation 3.2 gives a good indication of the range of allowed throughput (Obj. 1) to meet the downstream demand. However, this rule partially addresses the second part of the policy (Obj. 2) as it falls short if the competing direction of travels has equal opportunity. The l is fixed for all directions and is bound by local communication input and fixed storage capacity.

Rule 2: **Maximising speed gains v_{merg} for the exit flow at Z2 $S_{ext.Z2}$ and minimising difference-in-differences (DID) speeds for downstream discharge $Z3$ zones Δv_{Z3}.**

If a car accelerates a_{veh} from rest (0m/s) then $v_{ext} > v_{ent}$. This is true as speed $v = \frac{d}{t}$ and Z1 is assumed to be vacant from other road users. The Z2 is a section of the road where S_{ucl} merges with the passing S_{cl}. The traffic completely merges within l space of Z2 and before entering Z3. Hence, the release of S_{ucl} happens at two conditions: (i) within an acceptable gap time h allowed by incoming $S_{cl}>0$, and (ii) when $S_{cl} = 0$. The condition (ii) occurs only if the permitted p_{exe} is located at the S_{ucl} approach or when a turning movement is void ($V_{cl}=0veh$) during the permitted p_{exe}.

Overall, the ability of the exiting traffic stream S_{ext} to absorb the S_{ucl} without disturbance is determined by the available headway gap *h* and the road section's capacity C_{sec}. The C_{sec} drop happens due to bottleneck activation. It is observed that merging causes a reduction of speed on the main road depending on traffic composition (Laval et al., 2007). Therefore, if the merging occurs at Z2, then the difference in speed before and after merging at Z2 Δv_{Z2} is presented in equation 3.3.

$$\Delta v_{Z2} = v_{merg} - v_{ext} = \begin{cases} + & no\ disturbance\ to\ S_{ext} \\ 0 & v_{merg} = v_{ext} \\ - & disturbance\ to\ S_{ext} \end{cases} \quad (3.3)$$

If quantity $\Delta V_{Z2} \geq 0$ this presents that the section has acceptable *h* and suitable space in l_{Z2} and vice versa if $\Delta V_{Z2} < 0$. We call equation 3.3 maximising speed gains for exit flow S_{ext} at Z2.

The recovery Z3 is assumed to discharge traffic flow S_{dis} at nominal $v_{depart} > v_{merg}$ towards next junction and improve the capacity at Z2 to $C_{sec.Z2} < 1$. The $v_{depart} = v_{lmt}$ as this section of road has passed the merging zone and lane changing and/or overtaking (if any) behaviour towards the next junction is anticipated to take place beyond this zone. The difference Δv_{Z3} for each Z3 is detonated in equation 3.4.

$$\Delta v_{Z3} = v_{dept} - v_{merg} = \begin{cases} + & no\ disturbance\ to\ S_{dis} \\ 0 & v_{merg} == v_{lmt} \end{cases} \quad (3.4)$$

The Δv_{Z3} is used as a factor to weigh the impact of action p_{exe} on downstream discharge links. To maintain the system at equilibrium, the difference-in-differences *DID* between discharge zones Δv_{Z3} needs to be minimal, as in equation 3.5.

$$DID = min\Delta(\Delta v_{Z3}) \tag{3.5}$$

The *DID* determines the volatility of the executed action i.e., $p_{exe,}$ on the surrounding intersection discharge links in terms of speed *v* performance. This global observer factor accounts for all exit links. The lower the score of the *DID*, the better the action return. This is to ensure system integrity and assist in simultaneously monitoring exit zones.

The perseverance of Rules 1 and 2 is expected to meter upstream flow input, considering uncontrolled flow and downstream capacity. Rule 1, as in equation 3.2, maximises throughput to meet demand at exit links. On the other hand, Rule 2 tunes the input flow to ensure that the discharge links closely acquire travel speed across all intersection discharge links (equation 3.5) while maximising the speed gains in discharged flow (equation 3.3).

CHAPTER 4

METHODOLOGY

The flow chart of the methodology is presented in Figure 4.1. The process is divided into five (5) stages, including (i) data collection and processing, (ii) development of traffic micro-model environment, (iii) design of the DRL control system, (iv) training the DRL controller and (v) testing and analysis. Stages 2 to 4 are repeated for different traffic environment models.

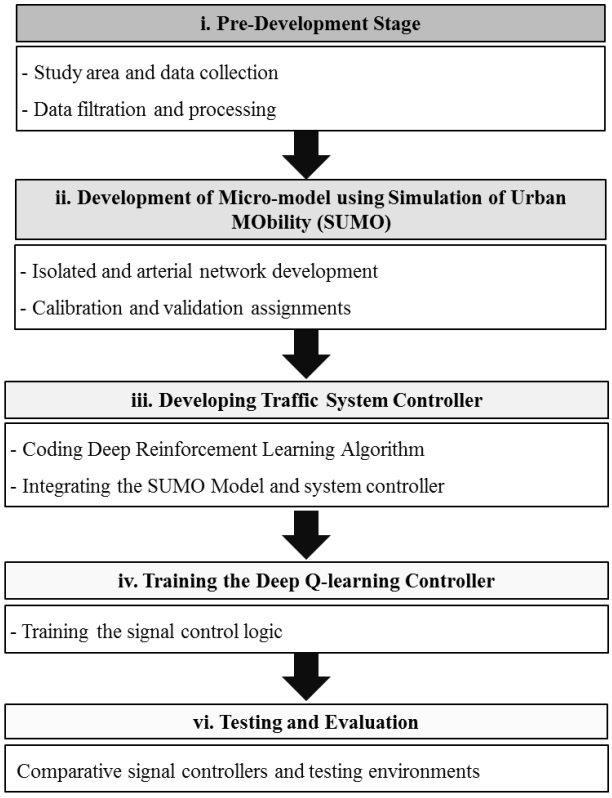


Figure 4.1: Flow chart for the study's methodology

4.1 Pre-development Stage

The data were obtained from a local traffic consultant company (JA Project Consultant Sdn. Bhd., Kuala Lumpur, Malaysia). The records reported that the traffic count survey was conducted on 5th September 2019. Based on the factsheet, the traffic survey was carried out during peak hours. The peak hour durations are between 7.00am to 10.00am and 4.00pm to 7:00pm. These hours represent the peak periods during which the highest daily traffic flows are encountered. The mobility vehicles were divided into five (5) categories: passenger car, small lorry (or van), heavy lorry, bus and motorcycle. The following Figure 4.2 shows the location of the study area.

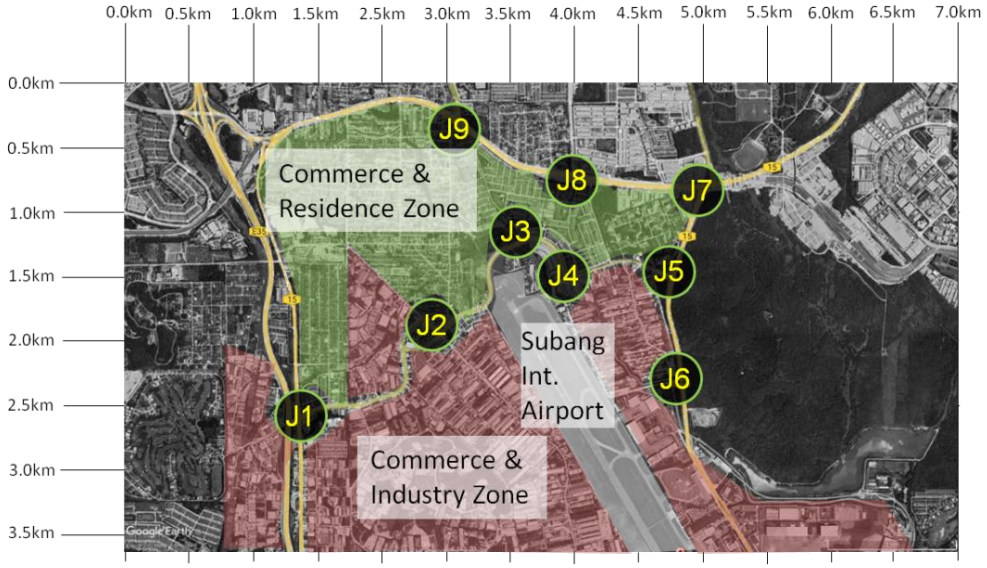


Figure 4.2: Study area and junction locations

The study region is a 3.5x5.5km^2 area in Subang Jaya, Selangor, Malaysia. The location accommodates land use types of residence, commerce, and industry. This mixed development site location makes traffic operation very challenging as road users' habits and journey purposes differ from one

class to another. Of further interest, the study region has an international airport, Sultan Abdul Aziz Shah Airport (Subang Int. Airport), within a 3km radius. There are nine (9) signalled intersections. These intersections are closely located near each other (≤1km distance). The only isolated intersection is J1, which is about 1.6km from J2. The isolated junction is as a signalled intersection with a distance of 1.6km or more from its closest neighbouring intersection (Manual, 2000). The longest travel distance is 7.5km, from J1 to J9. Table 4.1 presents the distance matrix between the junctions.

Table 4.1: Origin-Destination matrix based on measured distance (m) between the intersections

Junction ID	J1	J2	J3	J4	J5	J6	J7	J8	J9
J1		1,600	2,600	3,100	3,950	4,750	5,500	6,500	7,500
J2	1,450		1,000	1,500	2,350	3,150	3,900	4,900	5,900
J3	2,450	1,000		500	1,350	2,150	2,900	3,900	4,900
J4	2,950	1,500	500		850	1,650	2,400	3,400	4,400
J5	3,800	2,350	1,350	850		800	750	2,550	3,550
J6	4,600	3,150	2,150	1,650	800		750	1,750	2,750
J7	5,350	3,900	2,900	2,400	750	750		1,000	1,000
J8	6,350	4,900	3,900	3,400	2,550	1,750	1,000		1,000
J9	7,350	5,900	4,900	4,400	3,550	2,750	2,000	1,000	

The signalised junctions differ in terms of geometric configuration and road hierarchy. The number of controlled approach arms ranges from two (2) (as in J2) to four (4) (as in J1 and J7). Each junction layout has an exclusive slip lane for left-turn movement. Two (2) urban road categories in the illustrative case study include arterial and collector roads. The collector roads are between junction J1 and junction J5. The arterial corridors are between junctions J6 and J7, J7 and J9, and along the northbound and southbound travelling directions of junction J1.

The hourly traffic counts and vehicle composition are presented in Table 4.2. The passenger car class has the highest composition, with an average value of 67% of total road users. The second mode of transport is motorcycles, representing almost a quarter of road users. The medium lorry represents a 7% occupancy rate. The heavy and bus types of vehicles together account for about 2% of road users.

Table 4.2: Traffic counts and survey

Junction ID	Traffic Counts (veh/hr) per Peak Hour Count						6 Hours Vehicle Composition (%)**					
	7.00-8.00$^{C/L}$	8.00-9.00^{T}	9.00-10.00$^{V/I}$	16.00-17.00*	17.00-18.00^{T}	18.00-19.00^{T}	Passenger Car	Medium Lorry	Heavy Lorry	Bus	Motorcycle	Total
J1	5,639	5,439	4,213	4,085	4,970	5,308	71.00%	8.10%	1.90%	0.30%	18.80%	100%
J2	6,023	6,072	4,593	4,161	5,506	5,261	64.20%	7.00%	1.50%	0.30%	27.00%	100%
J3	4,859	4,677	3,389	3,240	4,175	4,134	66.80%	5.70%	1.30%	0.50%	25.90%	100%
J4	3,626	3,305	2,693	2,729	3,456	3,582	66.30%	5.60%	1.90%	0.50%	25.70%	100%
J5	5,919	5,477	3,894	4,320	4,920	4,794	66.70%	8.60%	2.50%	0.10%	22.20%	100%
J6	5,990	5,545	3,733	4,351	4,878	4,953	67.50%	7.70%	2.00%	0.30%	22.50%	100%
J7	6,670	6,505	4,799	5,004	5,663	6,069	69.30%	8.50%	2.10%	0.10%	20.00%	100%
J8	4,327	4,116	3,153	2,904	3,656	3,690	63.20%	5.40%	1.30%	0.04%	29.70%	100%
J9	4,995	4,427	3,125	3,120	4,317	4,858	65.00%	5.00%	0.90%	0.30%	28.90%	100%
Minimum Record	**3,626**	**3,305**	**2,693**	**2,729**	**3,456**	**3,582**	**63.20%**	**5.00%**	**0.90%**	**0.04%**	**18.80%**	**100%**
Maximum Record	**6,670**	**6,505**	**4,799**	**5,004**	**5,663**	**6,069**	**71.00%**	**8.60%**	**2.50%**	**0.50%**	**29.70%**	**100%**
Average Record	**5,339**	**5,063**	**3,732**	**3,768**	**4,616**	**4,739**	**66.67%**	**6.84%**	**1.71%**	**0.27%**	**24.52%**	**100%**
Total Record* **	**48,048**	**45,563**	**33,592**	**33,914**	**41,541**	**42,649**						

^{C}Calibration dataset, ^{V}Validation dataset, ^{L}Training DRL Logic, ^{T}Testing dataset, *Back-up dataset

****The hourly composition differs slightly from the 6-hours composition**

*****Unbalanced total traffic counts of all the intersections**

The junction capacity *C* under the fixed control was computed using formula 4.1 (Hunter-Zaworski et al., 2003).

$$C = (g/CT).S \quad (4.1)$$

Where *C* is the capacity of the signalised intersection (pcu/hr), g is the effective green time (s), *CT* is the cycle length (s), and *S* is the saturation flow rate (pcu/hr). The saturation flow rate of 1,900pcu/hr/lane is standard and frequently used (Hunter-Zaworski et al., 2003).

The region has a highly saturated flow rate during both peak periods. The utilisation of each junction varied notably, with some junctions experiencing medium flow (<66%) and others experiencing overflow (>100%). Table 4.3 presents the saturation flow and corresponding capacity for each junction. The computations indicate that the vicinity of the study area has a high saturated flow condition during the traffic count period.

Table 4.3: Existing capacity utilisation of the junctions at the study area

Junction ID	Saturation Flow (pcu*/hr) per Peak Hour Count						Capacity Utilisation (%) per Peak Hour Count					
	7.00-8.00	8.00-9.00	9.00-10.00	16.00-17.00	17.00-18.00	18.00-19.00	7.00-8.00	8.00-9.00	9.00-10.00	16.00-17.00	17.00-18.00	18.00-19.00
J1	3,212	3,237	2,790	2,867	3,221	3,373	81%	81%	70%	74%	83%	87%
J2	1,664	1,925	1,624	1,652	1,924	1,836	62%	72%	60%	58%	68%	64%
J3	3,180	3,146	2,803	2,777	3,279	3,154	78%	77%	69%	88%	104%	100%
J4	1,665	1,586	1,549	1,674	1,925	1,930	52%	49%	48%	51%	59%	59%
J5	2,148	1,842	1,783	2,054	2,178	2,112	62%	53%	51%	59%	63%	61%
J6	4,091	3,811	3,041	3,259	2,979	2,855	102%	95%	76%	84%	77%	74%
J7	5,795	4,202	3,614	4,066	4,080	4,770	126%	91%	78%	135%	135%	158%
J8	2,345	2,325	2,001	1,964	2,221	2,244	55%	55%	47%	46%	52%	53%
J9	2,679	2,392	2,011	2,123	2,680	2,758	83%	74%	62%	62%	78%	81%
Minimum Record	**1,664**	**1,586**	**1,549**	**1,652**	**1,924**	**1,836**	**52%**	**49%**	**47%**	**46%**	**52%**	**53%**
Maximum Record	**5,795**	**4,202**	**3,614**	**4,066**	**4,080**	**4,770**	**126%**	**95%**	**78%**	**135%**	**135%**	**158%**
Average Record	**2,975**	**2,718**	**2,357**	**2,493**	**2,721**	**2,781**	**78%**	**72%**	**63%**	**73%**	**80%**	**82%**
Total Record**	**26,779**	**24,466**	**21,216**	**22,436**	**24,487**	**25,032**						

***pcu is passenger car unit. The conversion factor is based on urban signal design conversion from the local consultant's traffic data, i.e., passenger car =1pcu, medium lorry = 1.19, heavy lorry = 2.27pcu, bus = 2.08pcu, and motorcycle =0.22**

****Unbalanced total saturation flow of all the intersections**

4.2 Development of Stochastic Traffic Micro-model

A reliable micro-model mimicking the actual traffic situation is required to understand the complex nature of the traffic network system. This simulation approach allows for modelling each vehicle explicitly. It is crucial to research multimodal traffic in the theory of traffic control (Wang et al., 2018). The Simulation of Urban Mobility (SUMO) software was used for this modelling mission.

The micro-model is developed using the Simulation of Urban Mobility (SUMO) software. The SUMO is an open-source, microscopic, multi-model traffic simulation software developed by the Institute of Transportation Systems at the German Aerospace Center in the year 2000. The advantage of SUMO is that it addresses various traffic model applications and has a Traffic Control Interface (TraCI) tool. The TraCI gives access to run a traffic simulation using an application programming interface (API). This feature is vital as it allows integration of the traffic control algorithm in stage C of this methodology. The API in this research is Python.

In order to achieve the objectives of the thesis, two traffic environments are developed, including (i) the isolated intersection J1 (4-leg) and (ii) the full network of nine (9) junctions. The traffic environment is defined as a 1-hour peak micro-model comprising five (5) classes of transportation modes. The configuration of each junction layout is presented in Figure 4.3.

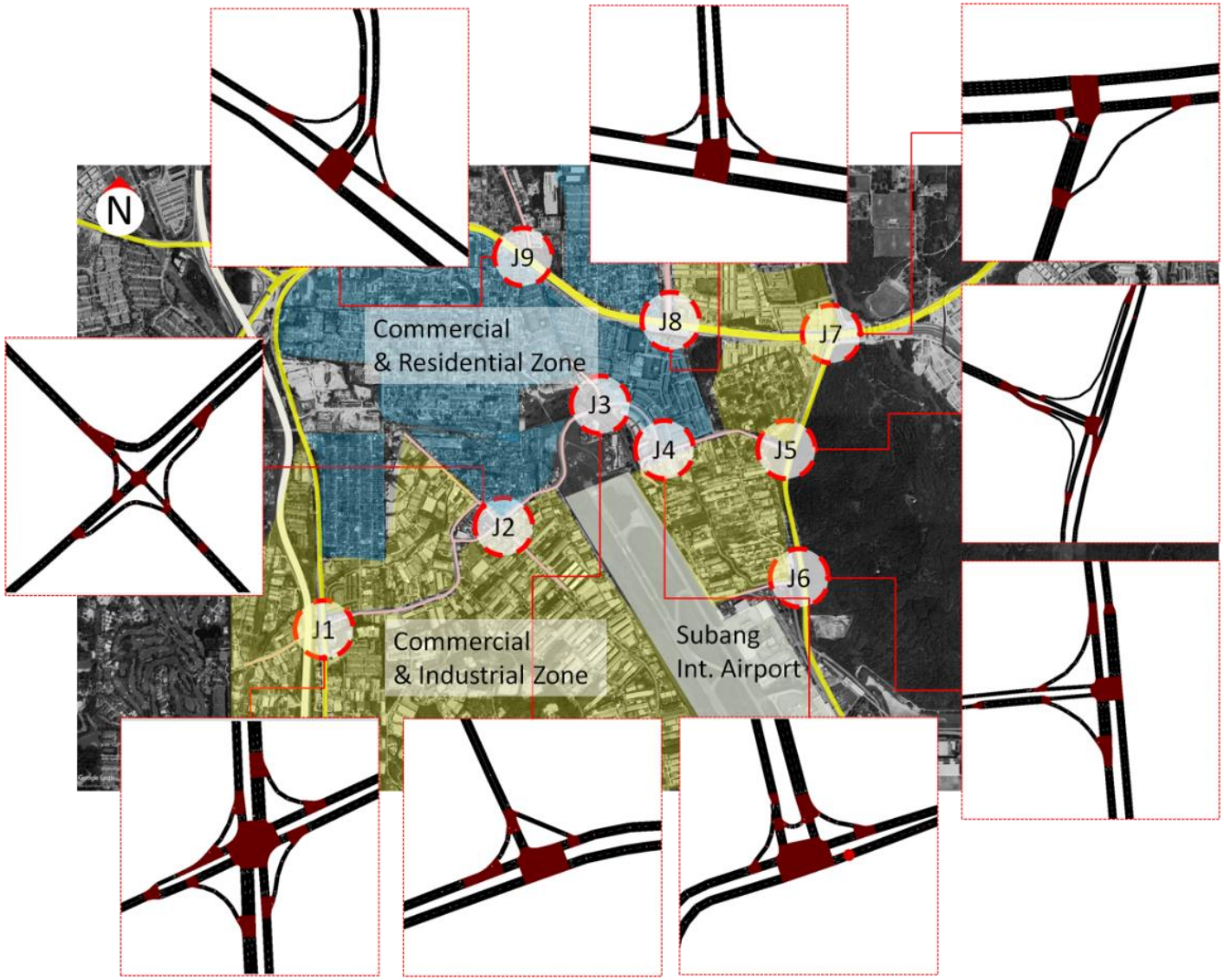


Figure 4.3: Study area and extracted junction layouts from the SUMO model

Isolated Junction Model: The purpose of the isolated junction model is to measure (i) the stochasticity of learning environment impact in stabilising adaptive traffic controller performance in various traffic conditions (objective 1), (ii) the significance of detection zone in developing an intelligent controller (objective 2), and (iii) the sustenance of the DRL agent under various traffic operation conditions.

Arterial Network Model: The goal of micro-modelling the arterial network is to (ii) scale single system design for network operation (objective 3), (i) test the sufficiency of local detection zones on network operation (objective 2), and (iii) verify the sustenance of the proposed downstream policy (Section 3.2) against other mainstream upstream policies in mitigating signal operation (objective 4).

The micro-modelling assignment is challenging, especially in capturing traffic flow dynamics. There are two (2) challenges related to modelling assignment, including (i) modelling uncertainty behaviour and (ii) identifying proper parameters to reflect driving conditions. The framework for building a model is presented in Figure 4.4.

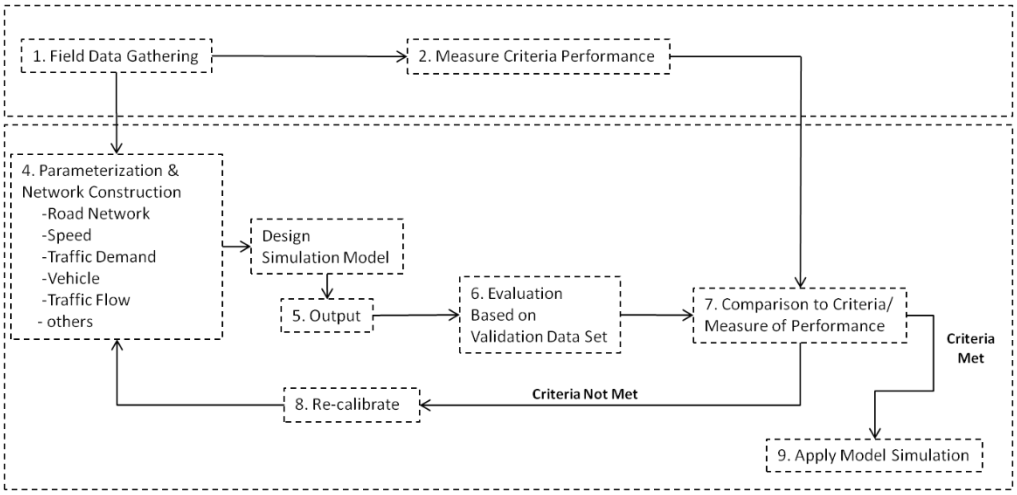


Figure 4.4: Calibration and validation procedure for micro-model

Integrated model assessment requires consistently dispersed data sources in a spatially and temporarily complete data set to provide the model inputs (Janssen et al., 2009). Recorded data usually lack quality and reliability as the details vary enormously in spatial and temporal dimensions regarding the site-exhibited behaviour. With increasing study area size, input data tend to be more uncertain relative to the point of the studied site (Kersebaum et al., 2015). Therefore, the uncertainty of the model increases simultaneously with the area under investigation (Antoniou et al., 2014).

Furthermore, if improper model parameters are used, a simulation model falls short of mimicking the field conditions (Park and Won, 2006). Therefore, there is a significant need to define the proper parameters (Antoniou et al., 2014). Parameterization estimates fixed model parameter values for single processes under controlled conditions.

These challenges create the need to perform the decisive calibration and validation procedure. Calibration is defined as adjusting model parameters outside the model code to fit their output to a set of measured state variables (Kersebaum et al., 2015). A measure of performance is applied to identify the performance of a certain parameter value. Once the best calibration parameter set is measured, the next step is to perform validation. Validation examines whether a model is not beyond its application and can describe other scenarios. This examination of a calibrated model should be against an independent data set that has not been used for calibration (De Wit, 1982). Validation is performed on an unused set of field data, such as different day conditions or field conditions.

4.2.1 Micro-Model Attributes

The SUMO includes numerous parameters that permit a user to define a traffic model. The modelling attributes are divided based on their functionality into four (4) models: vehicle model, car following model, lane changing model, and junction model. This division helps to identify a particular behaviour within the simulation environment. In total, 20 parameters and attributes fall within these categories. Figure 4.5 presents a summary of the four (4) categories of the micro-model.

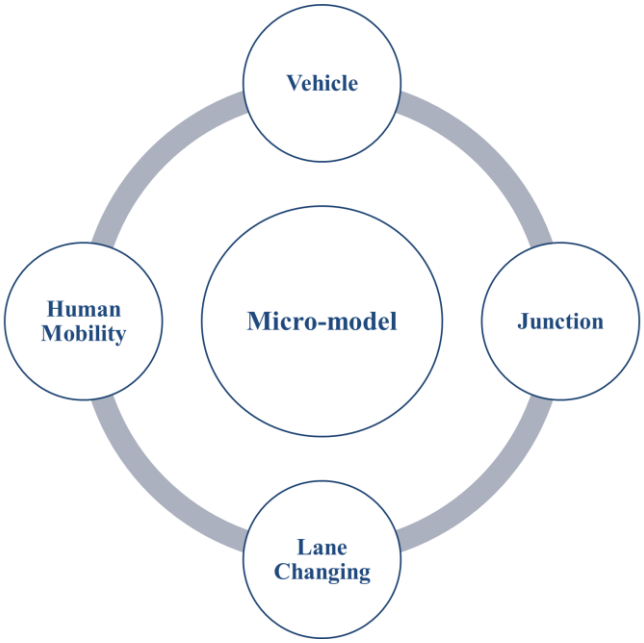


Figure 4.5: Micro-model and associated modelling attribute categories

4.2.1.1 Vehicle Model

The vehicle attributes control how it will be inserted into the network and how it leaves it. The details for each vehicle class dimension and default values as per the SUMO documentation manual are presented in Table 4.4.

Table 4.4: Vehicle class, physical dimension and speed features

Vehicle Class No.	Vehicle Type	Length (m)	Width (m)	Height (m)	minGap (m)	Accel (m/s^2)	decal (m/s^2)**	emergency decal. (m/s^2)	Max. Speed (km/hr)
1	Passenger	4.3	1.8	1.5	2.5	2.9	7.5	9	180
2	Motorcycle	2.2#	0.9#	1.5	2.5	6	10	10	200
3	Delivery	6.5	2.16	2.86	2.5	1.3	4	7	180
4	Trailer/Truck	12*	2.50*	3.5*	2.5	1.1	4	7	130
5	Bus	10**	2.50	3.4	2.5	1.2	4	7	85

#2m, 0.80m width (Malaysian market 99%)
*average of 12m is used to suit both trucks (7.1m) and trailers (16.1m)
**average of 10m is used to suit local buses of 8.5m and 12m, respectively

The SUMO has default values for these parameters. Nonetheless, some of the pre-defined values were changed to reflect the driving environment of the study area. For instance, the default maximum speed of 200km/hr (55.55m/s) was changed to 110km/hr (30.55m/s) as this is the maximum speed limit on Malaysian roads.

While vehicles were capped at the speed limit, individual speeds can vary to avoid homogeneous speeds and, consequently, invalid driving behaviour because vehicles cannot catch up with their leader vehicles. Assigning a speed factor is significant in capturing variation in cruise speed and making the simulation environment more realistic. The normal distribution was utilised to create the speed distribution using a 20% speed factor. The following Table 4.5 presents the vehicle attributes with default values.

Table 4.5: Vehicle Model Attributes

Parameter No.	Parameter Attribute ID	Default SUMO value	Remark
1	maxSpeed (m/s)	55.55	20.55
2	SpeedFactor	NA	95% vehicles drive between 80% and 120% of speed limit.
3	minGap (m)	2.5	
4	departPos (m)	base	
5	departLane	first	
6	departSpeed (m/s)	0	
7	maxSpeedLat	1	
8	latAlignment	center	

4.2.1.2 Car-Following Model

The car-following model is a replica of the driver's behaviour following another vehicle in the same lane in a traffic simulation (Kanagaraj et al., 2013). Despite enormous work in traffic flow theories, there is no such optimum traffic flow model to be accepted (Krauß, 1998). In this research, the Krauss theory is used as a traffic flow model. Krauss model performs well in non-steady or dynamic environmental conditions (Kanagaraj et al., 2013). This model is integrated into the SUMO software, and a modeller can customise two (2) attributes (sigma and tau) to calibrate the Krauss module. The sigma value controls a driver's imperfection. The tau value measures a driver's minimum desired time headway. Table 4.6 encloses the values of the Krauss model attributes.

Table 4.6: Car Following Model Attributes for Krauss Model

Parameter No.	Parameter Attribute ID	Default SUMO Value
1	Sigma	0.5
2	tau (s)	1.0

4.2.1.3 Lane-changing Model

The driving rules permit one vehicle per lane, but vehicles on the road intend to share the available lane space. This driving behaviour is most apparent among 2-wheeled vehicles. Two (2) observations were made from visual inspection: (i) motorcyclists drive in parallel to other vehicle classes on the road, and (ii) motorcyclists occupy the lane space of other road users

during stops or when trying to negotiate their ways in dense traffic conditions. The driving technique requires the formation of virtual lanes (sub-lane model) using the lane-changing model option, denoted as 'SL215', in SUMO software. By default, this model option is "off", and a modeller needs to activate the sub-lane model.

There are 10 parameters corresponding to the lane-changing model. These parameters measure various driver's behaviour, including strategic lane changing (lcStrategic), minimum lateral gap (minGapLat), eagerness to use the configured lateral alignment within the lane (lcSublane), willingness to accept lower front and rear gaps on the target lane (lcAssertive), maximum lateral acceleration per second (lcAccelLat), cooperative lane changing (lcCooperative), encroach laterally on other drivers (lcPushy), impatience (lcImpatience), reluctance to perform speed gains to place the vehicle across a lane boundary (lcLaneDiscipline), and probability to violate red light (jmDriveAfterRedTime). Table 4.7 summarises the default values of lane-changing parameters.

Table 4.7: Lane-changing Model "SL2015"

Parameter No.	Parameter Attribute ID	Default SUMO value	Remark
1	lcStrategic	1	
2	minGapLat (m)	0.60	
3	lcSublane	1	
4	lcAssertive (m)	1	
5	lcAccelLat (m/s^2)	1	1.80
6	lcCooperative	1	
7	lcPushy	0	
8	lcImpatience (s)	0 (no effect)	
9	lcLaneDiscipline	0	
10	jmDriveAfterRedTime	-1	0

4.2.1.4 Junction Model

The junction model is a representation of the geometric and lane configuration of the junctions. The layout was imported using OpenStreetMap. This tool is an available option for SUMO users. This was found to give better accuracy for the geometric design and road coordinates. Upon successful importation, the modeller must thoroughly check the intersection layouts and all associated links and nodes of the imported map. Moreover, it is necessary to ensure that traffic movements from a lane to a target lane are checked and verified. Wrong routing movement causes vehicle blockage and disturbs traffic insertion and flow.

Another part of the junction model is to include the signal programme. The junctions in the study area operate under a fixed controller system. From site data, it was observed that each junction had a different cycle plan and phase duration. Refer to Appendix A for detailed information about the time plan for each of the nine (9) signalised junctions. There are no measured values for junction model attributes.

After identifying and scripting the model attributes for the SUMO model and before performing the calibration and validation exercise, a preliminary exercise is required. This exercise aims to understand the modelling of software behaviour and evaluate the midnight threshold. The first step (a feasibility test) is crucial to determine the confidence level of the model's outputs and associated model iterations. The second step (a midnight

test) is a must to remove outputs that can skew the results from the model. Figure 4.6 charts the progress towards calibration and validation.

Figure 4.6: Flow chart of preliminary tests prior to calibration assignment

4.2.2 Feasibility Test and Minimum Simulation Run

The simulation uses multiple generating random numbers (RNG) to decouple different simulation aspects and reflect random behaviour. In this regard, SUMO implements the Mersenne Twister algorithm to generate seed value (SUMO Documentation). The default value in SUMO is fixed at 23423. Nonetheless, random RNG is activated throughout this research to ensure that the simulation model is stochastic and is not biased towards the deterministic behavioural output.

This stochastic modelling approach leads to variation in model environment dynamics and output. The solution is to conduct a feasibility test to determine the required number of simulation runs. This minimum number of iterations must achieve a certain level of confidence (CL) and accuracy in the model's stochastic behaviour and output.

The procedure for feasibility test is taken from Dowling et al. (2004). The minimum number of runs needs to fulfil the following equation 4.2.

$$C = 2 * \sigma * \frac{t_{(1-CI,N-1)}}{\sqrt{N}} \quad (4.2)$$

Where C is 1-CL, σ is the standard deviation for the simulation runs, t is the t-distribution value corresponding to the confidence interval and number of model runs, and N is the number of simulation runs. For this research study, a 95% CL is targeted. Table 4.8 presents the minimum simulation runs required to achieve the targeted CL for both developed models.

Table 4.8: Feasibility Test and minimum simulation run requirement

Model Type	Confidence Level (CL)	No. Of Simulation Runs (N)	Standard Deviations (σ)	$t_{(0.95,64)}$ Value	Confidence Factor (C)	Confidence Level to Standard Deviation (C/σ)
Isolated Junction	95%	64	49.35	1.998	24.66	0.50
Network	95%	10	236.14	2.262	338.00	1.43

4.2.3 Midnight Effect for Simulation Duration

How vehicles are inserted during simulation is significant to the model's overall performance. This is because, at the beginning of the simulation, the modelled section is empty, with no vehicles on the road. This initial part of the simulation is referred to as the 'midnight' driving period (Antoniuo et al., 2014). Vehicles during the 'midnight' period of the simulation do not experience external impacts of headways, speed restrictions, congestion, and travell time is perfect. Therefore, outputs from the start of the

simulation run have a strong tendency towards minor delays and do not reflect the actual traffic situation (Antoniuo et al., 2014).

In this research study, the identified midnight durations for the isolated intersection and network model are 66 seconds (1 minute and 6 seconds), and 780 seconds (13 minutes), respectively. The outputs from these initial durations are not considered for all assignment exercises, including calibration and validation (Stage 2), and testing (Stage 5). In addition, an extension of the peak hour model by midnight duration to have a complete 1-peak hour model was performed (i.e., 3660 seconds and 4380 seconds for isolated and network models, respectively).

4.2.4 Calibration and Validation of Micro-model Traffic Environment

The challenge with micro-model assignment is the ability of a model to represent traffic conditions. A simulation model falls short of mimicking the field conditions if improper model parameters are used (Park and Won, 2006). The calibration and validation is an iterative process until the model parameters are adjusted within a reasonable range. The calibrated attributes are categorised based on their functionality into three (3) models: vehicle, car following, and lane changing. This division helps to identify a particular behaviour within the simulation environment. A performance measure is applied to identify the performance of certain parameter values. Figure 4.7 presents the cycle of calibration and validation assignment.

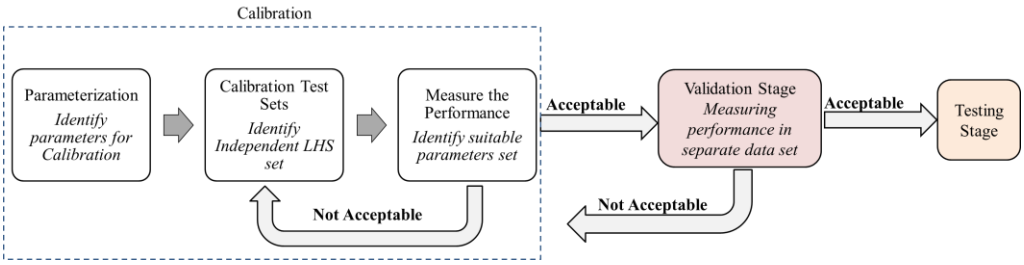


Figure 4.7: Procedure steps for calibration to testing

The parameterization estimates fixed model parameter values for single processes under controlled conditions (Kersebaum et al., 2015). As five (5) classes of vehicles represent the heterogeneous driving environment, some of the calibrated attributes differ per class user. Once the parameters are tuned using the calibration test sets, validation is performed. Validation examines whether a model is not beyond its application and can describe other scenarios. This examination of a calibrated model should be against an independent data set that has not been used for calibration (De Wit, 1982). To validate a model, the appropriate measure of performance is needed.

4.2.4.1 Measure of Performance (MoP)

The measure of performance (MoP) characterises the distance between the aggregate measurements observed from the real traffic data and the simulation results (Zhang et al., 2008). The most commonly utilised measurements are link counts from various network locations, average travel speed, and trip travel time (Zhang et al., 2008). To measure the simulated traffic counts, detector loops were placed at the end of each approach lane (before the stop line).

Several fitness functions are commonly utilised as a measure of performance. Herein, a statistic called Geoffrey E. Havers (GEH) is used for the calibration target and goodness of fit. The GEH formula is helpful in creating a mathematically consistent data set that can be used for travel demand forecasting and traffic simulation models. The GEH is useful in comparing two (2) different flow values using the following equation 4.3.

$$GEH = \sqrt{\frac{(V_2 - V_1)^2}{(0.5(V_1 + V_2)}} \quad (4.3)$$

Where V_1 is the observed vehicle count from the traffic survey, and V_2 is the modelled vehicle count from the simulation run.

As a rule of thumb, in comparing assigned and observed volumes, a GEH parameter of 5 or less is considered acceptable, and links > 10 GEH would require closer attention (Horowitz et al., 2014).

It is important to mention that the GEH was computed for controlled lanes. Therefore, a second MoP is appended to represent the overall model accuracy. The environment model is of good quality if its output is within a 15% margin of error to the site condition (Dowling et al., 2014).

Examining the validation sets for both micro-models, the MoP outcome based on 95% CL indicates that the isolated micro-model intersection has 94.05% accuracy and a GEH value of 2.03. The network model has also achieved acceptable accuracy at 93.00% for link counts, 87.50% for travel

time and speed, and a 1.46 GEH score. Table 4.9 presents a summary of the MoP attributes for both developed models in this study.

Table 4.9: validation summary with the MoP attributes

Validation Set	MoP Attribute			
	Link Count (veh)	**Travel Time (sec)**	**Speed (m/sec)**	**GEH**
Site Data	5,301	NA	NA	
Isolated Junction Model	4,985	NA	NA	
Difference (Model:Site)	94.05%	-	-	2.03
Validation Set	**Link Count (veh)***	**Travel Time (sec)***	**Speed (m/sec)***	**GEH**
Site Data	23,099	129	11.22	
Network Model	22,811	151	10.13	
Difference (Model:Site)	93.00%	87.50%	87.50%	1.46

*Based on link counts ≤15%error. Refer to Appendix B for further details

This study modelled the isolated intersection first; its calibrated values were incorporated into the network's micro-model. The calibration and validation assignment for the network indicated that all values are suitable except for the maximum speed and departure position of a vehicle. The maximum speed for the network was reduced to 20.55m/s. This speed value is suitable as the arterial road network has a design speed limit of 70km/hr. The departure location of vehicles was "freed". Therefore, vehicles can be inserted in the most suitable position at the insertion link. This finding is suitable as the network model's origin-destination trip is much more complex than the isolated intersection. The summary list of these parameters and their calibrated and validated values are presented in Table 4.10.

Appendix B presents detailed calibration and validation procedures for the isolated intersection and network models.

Table 4.10: Calibrated and validated modelling attributes

Parameter No.	Parameter Attribute ID	Description	Default SUMO value	Calibrated Value Range	Validated Value
i.	**Vehicle Model Attributes**				
1	maxSpeed (m/s)	Maximum velocity	55.55	30.55 (fixed for all)	30.55 (isolated), 20.55(network)
2	SpeedFactor	Speed multiplier to vary the speed among vehicle of fleets		95% vehicles drive between 80% and 120% of speed limit. (fixed for all)	Normc(1,0.1,0.2,2)
3	minGap (m)	Empty space after leading vehicle during stopping	2.5	0.80-3.2	Passenger car and motorcycle: 0.89 Medium lorry: 1.07 Heavy lorry and bus: 2.50
4	departPos (m)	Insertion position for vehicle to enter the network	base	“>=0”, "random", "free", "random_free", "base", "last", "stop"	Position (isolated), free (network)
5	departLane	Departing lane of inserted vehicle	first	≥0, "random", "free", "allowed", "best", "first	Free
6	departSpeed (m/s)	Initial speed of inserted vehicle to network	0	≥0, "random", "max", "desired", "speedLimit"	Max
7	maxSpeedLat	Maximum lateral speed	1	>=0	1.80
8	latAlignment	Preferred lateral alignment for the sublane-model.	left, right, center, compact, nice, arbitrary	Motorcycle: “nice” Other classes: “center”	Motorcycle: “nice” Other classes: “center”
ii.	**Car Following Model (Krauss model theory)**				
9	Sigma	Driver’s imperfection (0 denotes perfect driving)	0.5	0-1	Passenger car and motorcycle: 0.22 Medium lorry: 0.44 Heavy lorry and bus: 0.50
10	tau (s)	Driver's desired (minimum) time headway	1.0	1.0-3.0	1.0
iii.	**Lane Changing Model: SL2015**				
11	lcStrategic	Eagerness for performing strategic lane changing	1	0-4	Passenger car and motorcycle: 0.67 Medium lorry: 1.33 Heavy lorry and bus: 1.00
22	minGapLat (m)	Desired minimum lateral gap when using the sublane-model	0.60	0.15-2.5	Passenger car, motorcycle and medium lorry: 0.67 Heavy lorry and bus: 0.60
13	lcSublane	Eagerness for using configured lateral alignment within a lane	1	0-4	Passenger car, motorcycle and medium lorry: 4.00 Heavy lorry and bus: 1.00
14	lcAssertive (m)	Willingness to accept lower front and rear gaps on the target lane. The gap is divided by this value	1	0.15-2.5	Passenger car: 0.41 Motorcycle: 0.33 Medium lorry: 0.41 Heavy lorry and bus: 1.00
15	lcAccelLat (m/s^2)	Maximum lateral acceleration	1	1.80	1.80
16	lcCooperative	Willingness for performing cooperative lane changing. Lower values result in reduced cooperation	1	0-1	Passenger Car and Motorcycle: 0.94 Medium lorry: 0.89 Heavy lorry and bus: 1.00
17	lcPushy	Willingness to encroach laterally on other drivers	0	0-1	Passenger car and motorcycle: 0.83 Medium lorry: 0.67 Heavy lorry and bus: 0.00
18	lcImpatience (s)	Dynamic factor for modifying lcAssertive and lcPushy.	0 (no effect)	-1,0 to 1	Passenger car, motorcycle and medium lorry: 0.44 Heavy lorry and bus: 0
19	lcLaneDiscipline	Reluctance to perform speedGain-changes that would place the vehicle across a lane boundary	0	0-4	Passenger car and motorcycle: 0.22 Medium lorry: 0.44 Heavy lorry and bus: 0
20	jmDriveAfterRedTime	Driving at yellow light and break at red.	-1	0	0.00

4.3 Traffic Signal Controller Development

As stated in Section 3.1, this research study proposes a single system controller based on a deep Q-leaning algorithm. The signal operation is independent of other neighbouring controllers. This system concept decentralises decision-making to the local intersection only.

Hidden layers between input and output characterise the deep Q-neural technique. This technique is capable of classifying nonlinear data and feature extraction compared to other machine learning types where boundary classification is limited to linear data.

The task of interference between the agent and the environment includes decision-making (action), a policy to guide the action process, and a reward to evaluate the choices. Ultimately, the agent is expected to reach an optimal operation that maximises the reward's return. The interaction between the agent controller and the traffic environment is depicted in Figure 4.8.

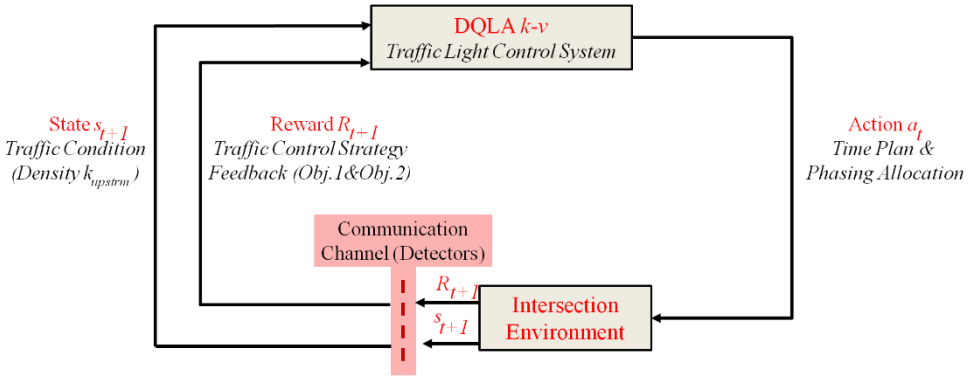


Figure 4.8: Adaptive controller interaction with the traffic environment

Based on Figure 4.8, at time step; *t*, the agent controller observes an input or state; s_t. Based on this input, the agent selects an appropriate traffic signal phase; output, this choice is referred to as action; a_t. As vehicles move under this action phase a_t, a new state, s_{t+1} is received. The performance of the action, a_t, is measured, and the agent receives a reward, R_t at the end of the time step; *t+1*. In time progression, the learning agent engages with the intersection utilising the reward function as a guide to make the decision and to move towards an optimal solution that maximises its rewards or decreases its punishment.

4.3.1 Approximation Technique for Developed Controller Agent

The Q-learning algorithm for a pair of state *s* and action *a Q(s,a)* and a reward *R* in a timestep *t* is represented by the Bellman equation (Wang et al., 2018) in equation 4.4.

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(R_{t+1} + \gamma Q_{max_a}(s_{t+1}, a) - Q\ (s_t, a_t)) \qquad (4.4)$$

To accommodate future rewards, a discount rate γ is commonly used. The weighting factor α adds significance to the future. Further explanation for these factors is in Section 4.4 Training and Testing.

Two (2) traffic signal systems were developed in this study. The main difference between these logic systems is the traffic control policy (reward).

1. **Logic 1: Deep convolution neural network (DCNN) agent**. The DCNN is in line with other adaptive controller studies. The logic programme utilises an upstream control policy to operate a signal intersection. The development of this logic system is to study the impact of the stochastic traffic environment on learning and testing the controller and the application of enclosing loop detectors with defined detection areas (local protocol). Further specifics about the DCNN system are presented in Section 4.3.2.
2. **Logic 2: Deep sequential Q-learning agent based on density-speed policy at discharge routes (DQLA *k-v*)**. The QLA *k-v* is the first adaptive controller to utilise downstream conditions to control a signalised junction. This development aspect aims to test the efficiency of the novel downstream policy in mitigating arterial network operation and validate the capacity of decentralised adaptive single controllers in a network context using a local communication protocol. Further details about the DQLA *k-v* system are presented in Section 4.3.3.

4.3.2 Upstream Controller: DCNN Agent

4.3.2.1 State Inputs

The DCNN logic observes using discrete lane cells (DLC). There are three (3) inputs from the traffic environment, including (i) the speed of a vehicle, (ii) the position of a vehicle, and (iii) signal phasing. These inputs correspond to vehicle information and junction state. The observations are registered within a time interval *t* equivalent to green phase duration T_{green}.

The input shape for these parameters depends on the number of lanes and edge configuration. The input matrices should propagate for a specific stretch of road using a reference point. In this regard, a 70 metres catchment is chosen. This choice is found to be practical, as existing hardware devices are capable of covering such a detection area. Lanearea detectors are used in the SUMO model of the environment. Lanearea detector captures the traffic within a specified area along a lane.

The traffic light status is either green or yellow. The yellow phasing is indicated with "1", while "0" represents the green phase. The following matrices in equation 4.5 represents the state input for the speed, position, and traffic light.

$$Position = \begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_{70} \end{bmatrix} \quad Speed = \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_{70} \end{bmatrix} \quad Traffic\ Light = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{4.5}$$

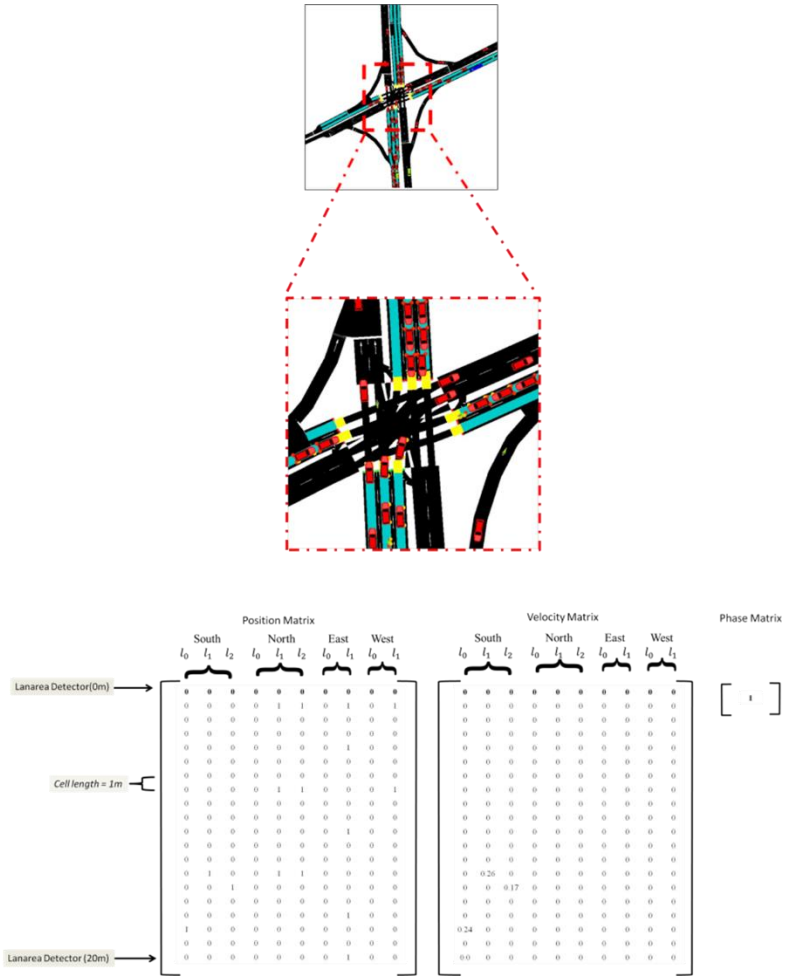


Figure 4.9: Explanatory snapshot at time step *t* of the isolated intersection and two input matrices for position and velocity for a 20 metres length of road from stop lines as received by the DCNN agent

Table 4.11 summarises the neural structure of the DCNN logic.

Table 4.11: Parameters for the DCNN agent

Agent	**DQLA *k-v***
Hidden Layer Class	Conv2D
Number of Hidden Layers	3
Activation Function	ReLU
Loss Function	Mean Squared Error (mse)
Target Network Function	Gradient Desecnt (RMSprop)
Weight Factor for Loss Fucntion, *tau*	0.15
Output Layer Class	Flat
Activation Function-Output Layer	Linear

4.3.2.2 Traffic Control Reward Policy

The reward is a traffic control strategy utilized by the agent to understand the effects of the latest action (Nagabandi et al., 2018). The reward strategy R_t for DCNN agent is cumulative halting time for vehicles during each action. This R_t is vector value with negative or positive arithmetic. The positive reward ($R_{t+1} > 0$) points out a decrease in delay. The negative reward ($R_{t+1} < 0$) relates to an increment in halting time. Hence, a phase re-allocation will be executed if the halting time in the green direction falls below the cumulative halting time in other directions, as represented in equation 4.6.

$$R = R_G - \sum R_R \tag{4.6}$$

Where R is the total value of vector reward responding to the summation of halting vehicles during the green phase R_G and red phase R_R.

4.3.3 Downstream Controller: DQLA *k-v* Agent

4.3.3.1 State Input

The DQLA *k-v* logic observes a 1-dimensional input. The state input is the density in each direction of travel. Like the DCNN controller, a detector observes the state within 70 metres from a stop line. The SUMO model can extract the required value within the specified communication detection zone.

This observation is registered within a time interval *t* equivalent to green phase duration T_{green}.

$$Input = [k_{1,j}]1 \leq j \leq N \quad (4.7)$$

Where *k* is the average density per the direction of travel, and *N* is the number of junction approaches.

Table 4.12 summarised the structure of the DQLA *k-v* logic.

Table 4.12: Parameters for the DQLA *k-v* agent

Agent	DQLA *k-v*
Hidden Layer Class	Dense
Number of Hidden Layers	2
Activation Function	ReLU
Loss Function	Mean Squared Error (mse)
Target Network Function	Gradient Desecnt (RMSprop)
Weight Factor for Loss Fucntion, *tau*	0.15
Output Layer Class	Dense
Activation Function-Output Layer	Softmax

4.3.3.2 Traffic Control Reward Policy

The intelligent DQLA manages signal operation based on the proposed downstream *k-v* traffic control policy, as presented in Section 3.2. The reward *R* represents the *k-v* components as in equation 4.8.

$$R = R_k + R_s \quad (4.8)$$

The scalar reward R_k is a ratio of the downstream density to the optimum density at Z3 (k_{Z3}:$k_{op.dwn}$). To generalise favoured density values, the density ratio R_k is categorised into five levels (close definition to Level of Service LOS for road capacity) as in Figure 4.10. The highest positive reward is given for near optimum condition ($k_{Z3} \approx k_{op.dwn}$). The optimum k_{Z3} is a 15% (above or below) k_{op} value. The award R_k deteriorates as k_{Z3} records value further away from $k_{op.dwn}$. A punishment is awarded if a p_{exe} leads to k_{Z3} =0. Because of incurred waiting time costs associated with the poor allocation of the assigned signal phase.

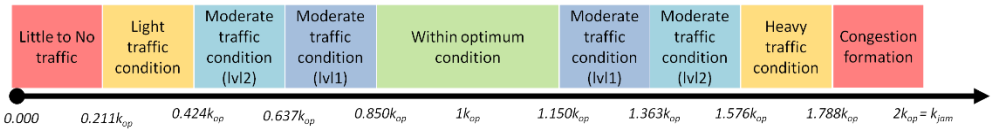


Figure 4.10 Categorical reward return for density to optimum density ratio at discharge zone

The $k_{op.dwn}$ is half of jam density k_{jam}. The k_{jam} is computed from the Greenshield model (1934) given the road section capacity k, average speed v, and free-flow speed v_f as in equation 4.9. The Greenshield model was found to be the best-fit model to represent the speed-density relationship (Khoo and Tang, 2016).

$$k_{jam} = \frac{k}{\left(1 - \frac{v}{v_f}\right)} \tag{4.9}$$

The $k_{op.dwn}$ is as in the following equation 4.10.

$$k_{op.dwn} = \frac{1}{2}\, k_{jam} = \frac{k}{2\left(1-\frac{v}{v_f}\right)} \tag{4.10}$$

The scalar R_s in earlier equation 4.8 consists of two (2) quantities: (i) a reward related to maximising speed gains for exit flow $S_{ext.Z2}$ at Z2 as in earlier equation 3.2 (Section 3.2), and (ii) a reward factor to quantify the *DID* for speed *v* performance as in earlier equation 3.5 (Section 3.2). To scale the *DID*, a coefficient of variation *CV* is utilised. The *CV* value is the ratio of standard deviation σ of the speed difference Δv_{Z3} at discharge links affected by p_{exe} to the mean μ of the speed difference Δv_{Z3} at unaffected discharge links. The lower the degree of the *CV*, the better the action return. A negative operator is added as *v* is not a negative value and to punish the agent if the ratio is large, as in equation 4.11.

$$CV = -\frac{\sigma}{\mu} \tag{4.11}$$

4.3.4 Action Assignment and Phase Control

The traffic light operation is cycle free. The acyclic control gives complete freedom to the agent to decide on phase allocation and duration assignment. The agent decides on the time plan and phase duration. Each action should ensure a transition from green to yellow and vice versa. A governing duration of time will last before the agent is allowed to implement a new phase plan. Each execution of an action A_t will undergo a transition process in the following order:

1. Changing a green phase into a yellow phase.
2. Turning the yellow phase into a red phase.
3. Assigning a new green phase.

The number of permissible actions a is equivalent to the number of competing traffic movements X i.e., ($a = X$). The concept of the phasing plan is to dedicate a single phase for each approach to avoid conflicting movements, as in Figure 4.11.

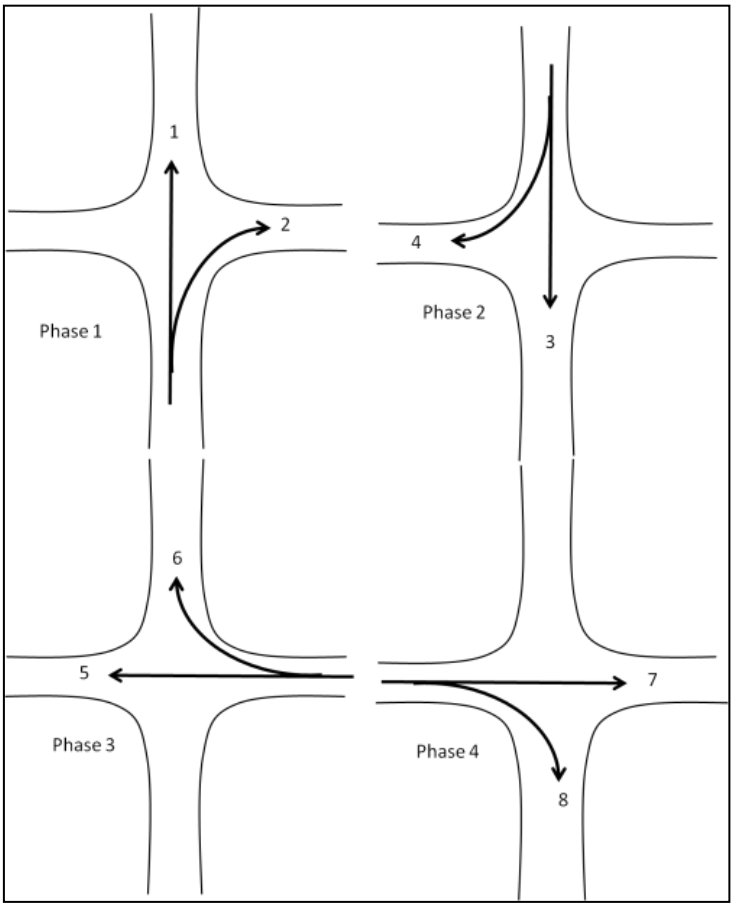


Figure 4.11: Executive phase action for the intelligent controller

4.3.4.1 Phase Timing for Isolated Intersection Model

One of the isolated testing model's main objectives is to investigate the environment model's extension in DRL and the ability of the adaptive controller to act effectively in the traffic environment. Earlier studies in DRL indicated unrealistic phase timing (<5 seconds). The naive timing durations not only skew results in favour of the proposed DRL but also negatively impact and jeopardise the safe operation.

In order to reduce such bias, the DCNN signal controller executes an action every 20 seconds. The action timing corresponds to 16 seconds of green time (T_{green}) and 4 seconds of yellow time (T_{yellow}). The green allocation is closely related to the recommended minimum green phase time of 15 seconds to match the driver's expectation (Urbanik et al., 2015). In addition, the phase is benchmarked closely to the least phase of the present fixed signal controller during the survey data collection. The least phase duration is 20 seconds from site records.

4.3.4.2 Phase Timing for Network Model

In the network model context, a dynamic phase time is used. The effective green time comprises green time T_{green} and yellow time T_{yellow}. The G_{eff} ranges between $T_{min} \leq G_{eff} \leq T_{max}$. Whereas, the T_{yellow} is fixed at 4 seconds, the T_{green} is flexible between minimum $T_{min.green}$ and maximum $T_{max.green}$ durations.

Minimum duration $T_{min.green}$: The T_{min} is required to meet safe traffic operations. Two (2) timing attributes are associated with this requirement, including (i) start-up lost time T_{lst} of 2 seconds (Manual, 2000), and (ii) timing required to reach optimum flow rate $T_{opt.min}$. The $T_{opt.min}$ is equivalent to time required for four (4) vehicles to pass an intersection. To pass the intersection, the headway gap *h* is computed using the kinematic law incorporating conflict zone length *s*, vehicle's acceleration *a*, and number of successive cars left the intersection *O* as in equation 4.12.

$$h_i = \frac{\sqrt{(-a.\frac{1}{s_m}.(O-1))^2+2.a.s}}{a} \tag{4.12}$$

Maximum duration $T_{max.green}$: The T_{max} is computed from the last vehicle moving from rest a_{last} at the end of *l* as in equation 4.13.

$$T_{max} = \sqrt{\frac{2l}{a_{last}}} \tag{4.13}$$

4.4 Training the Deep Q-learning Controller

Solving the Bellman function (earlier equation 4.4) to find the optimal policy for the agent control requires a comprehensive search for all scenarios, forecasting their chances of occurrence, and evaluating their profitability in terms of expected rewards. Hence, finding the optimal policy will require two (2) elements: (i) a complete understanding of the environmental dynamics and

(ii) a computational capacity and hardware resources capable of providing complete computation of the solution (Sutton and Barto, 2018).

Two (2) challenges were encountered in training the proposed agents. First, there is no documented guideline for training a DRL agent for traffic signal control, specifically for the dynamic and stochastic environment, as in this study. Second, the model of arterial network training is more challenging and requires more computational power compared to the isolated intersection. Therefore, we took the initiative to document a systematic training procedure for the DRL agent. Figure 4.12 presents a flow chart for the DRL training.

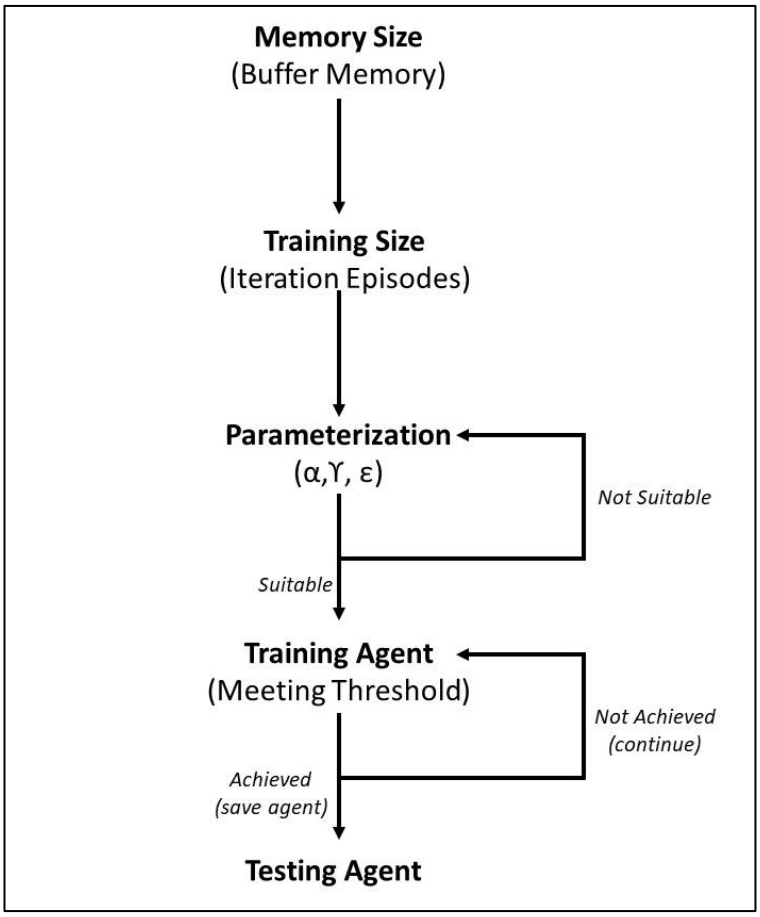


Figure 4.12: Flow chart for DRL Training

4.4.1 Pre-training Agent

This stage is essential to determine the minimum requirements for memory size, iteration runs, and grid search (if required).

4.4.1.1 Replay Memory Size

The training is offline, where the system's memory and neural network are created. Pre-training is required to accumulate tuples of states, actions, rewards, and experiences for replay memory. Initially, the decision on action is random, where the agent selects an element from the specified action range. The random selection follows the normal (Gaussian) distribution. The Gauss distribution is a continuous probability distribution for a real-valued random variable that takes the form of a density function, as in the following equation 4.14.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}}\, e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \quad (4.14)$$

Where f(x) is the distribution function, σ is the standard deviation of observations, μ is the mean of observations, and *x* is an examined observation in the data.

There is no specific optimal memory size. Studies in traffic signal controls reported various ranges of memory sizes for their training assignments. A study by Liu and Zou (2018) showed that buffer size substantially affects an agent's learning dynamics. Too much or too little memory can slow down the value function learning. In this aspect, we recommend that a reply memory size be at least sufficient to

accommodate one (1) complete model run assignment incorporating the warm-up period. The phase time definition determines how many instances or actions are taken. This threshold value is computed using the following formula 4.15.

$$\#instances\ (per\ episode) = \frac{simulation\ time}{action\ phase} \tag{4.15}$$

For the isolated model, the number of instances per episode amounts to 183 or (3660 seconds/20 seconds). For the network model, the number of instances per episode amounts to 486 or (4380 seconds/9 seconds). The instant is equivalent to one (1) action decision per agent. Setting a higher memory size could capture higher CL on the model and more instances of the environment. For the arterial traffic network, little more than two (2) hours (2 training episodes) of environment instances were embedded into the memory at 1,200 memory size. The isolated intersection model had a buffer size of 2,000, equivalent to 11 hours of operation instances. These values were also suitable for the available computing power.

4.4.1.2 Iteration and Episode Runs

After the memory is filled with random actions, the agent begins training. The new experiences are embedded in the memory to replace the older decisions. The training is carried out in a continuous task involving episodes. Each episode is equivalent to one (1) model run (i.e., peak hour simulation time). The least number of episodes during the training is determined from traffic model features. The stochastic approach in this study requires the model to be freed from the RNG value. In addition, the traffic flow is dynamic, with a 20% variation. In other words, each simulation episode is different from other episodes.

To ensure that the trained agent captures environment dynamics accurately, the least number of training episodes needs to follow the 95% CL threshold (Section 4.2.2). The CL means that the re-occurrence of a particular dynamic happens within a 95% chance in these iterations. So, the training should comprise at least 64 episodes for the isolated intersection model and at least 10 episodes for the network model. Otherwise, the trained agent will fall short of sufficient knowledge of the environment.

The number of training episodes is not to be confused with the pre-training episodes (build-up of buffer memory). The summation of both episodes will give the minimum iteration runs required to build up memory and then train the DRL to achieve optimal value function.

4.4.1.3 Hyper-parameter Tuning

The greedy Q-Learning policy π selects alternatives based on immediate and local considerations without considering the long-term alternatives that could be better present decisions (Sutton and Barto, 2018). The π policy takes the following form in equation 4.16.

$$\pi = arg\max Q(s,a) \text{ for all } s \in S, a \in A \tag{4.16}$$

Choosing the correct value for parameters is significant for training the model more effectively. Two (2) parameters impact the Q-learning algorithm, as in earlier equation 4.4. These parameters are (i) the learning rate $\propto$ to

control the action-value assessment and (ii) the discount factor γ to weigh short-term (immediate) and long-term rewards. The γ is crucial to give significance to the future rewards over the immediate ones. Both of these factors are $\in[0,1]$. Once the Q-function is estimated, the agent selects an action reflecting the highest value paired with a present state.

However, the agent can get stuck in local minima if no proper exploration strategy is in place (Sutton and Barto, 2018). Therefore, to balance the exploration and exploitation approaches, a trade-off decay ε factor is implemented where $0 \leq \varepsilon \leq 1$ (Mannion et al., 2016). At the value of 1, exploration is chosen. On the other hand, with probability 1-ε, exploitation is chosen.

To begin with, there are popular default values commonly used in literature studies. For instance, the default values for α, γ and ε are 0.001, 0.95, and 0.995, respectively. While these values were found suitable for training the DRL agent in the isolated intersection model, they proved very costly to train the agent in the arterial network condition. In particular, the decay factor ε was very costly for the network micro-model.

In order to decay from 1 (exploration) to 0.01 (exploitation), 198 model runs will be needed. These runs caused the training assignment of the 14,000 vehicles network to cripple as it exceeded the computation power of the used machine. In addition, the SUMO software is a single-thread CPU. Hence, running parallel multi-thread sessions was not applicable to reduce the

load on the computer. Several solutions were also attempted, such as using remote servers and cloud computing machines. However, the most practical solution is to perform a hyper-parameter exercise. This solution is practical when computation power is restricted.

The grid search exercise yielded attribute values of 0.001, 0.50, and 0.44 for the epsilon, learning rate, and weight, which are most suitable for the DRL algorithm. Further details on the hyper-parameter tuning are detailed in Appendix C.

4.4.2 Training Agent

The training was carried out using a Windows 7 professional 64-bit operating system with process specifications of Intel(R) Xeon(R) CPU E5-1650 @ 3.20GHz and Random-access memory (RAM) of 32.0GB.

The training for the isolated intersection included a total of 500 episodes. Each episode is a simulation run of 63 minutes, or about 22 days of continuous traffic. The agent's learning assignment took little more than 24 hours (one day) to complete the training session on the mentioned machine.

The arterial network learning process included 100 to 180 episodes for DQLA *k-v* and DCNN agents, respectively. Each episode took an average of three and a half (3.5) hours to complete. Overall, it took about 16 days to complete the training session for DQLA k-v and nearly 23 days to complete the training of DCNN agent for the network environment due to the machine's capacity.

Table 4.13 summarises the training attributes for DRL agents in the isolated and network traffic micro-models.

Table 4.13: Agent training and environment model

Model	Isolated	Network
Traffic Volume (veh/hr)	5,639	14,182[B]
Memory Size	2000	1200
Batch Size	32	128
Minimum Episodes[#]	273	15
Actual Training Episodes	500	100 (DQLA *k-v*) and 180 (DCNN)

Model	Isolated	Network
Discount Factor α	0.95	0.50
Learning Rate γ	0.001	0.001
Decay Factor ε	0.995	0.44
Training Duration (hrs)	84	140
Number of Trained Instances*	91,500	437,400 (DQLA *k*-*v*) ** and 787,320 (DCNN)**

[B]Reported volume after balancing the network of 48,048veh/hr
[#]Estimated from the required episodes for memory size+95%CL+Exploration-exploitation episodes
*Estimated from the product of the actual training episodes and the number of instances per episode
**Total of all nine (9) intersections of the arterial network

4.4.2.1 Performance Measure

The traffic signal problem represents the NP-hardness class (Al Islam and Hajbabaie, 2017). The properties of the solution are not linear, making the convergence to the solution a challenge for the agent. The convergence problem is faced in this stochastic, highly dense training environment. Similarly, it is not anticipated that the agent during training will not achieve optimal decisions at every instant. The dynamics of traffic change with every decision. Nevertheless, the agent must enhance operation within the specified traffic flow duration (i.e., peak hour).

Therefore, a performance measure is used to identify the right agent for testing. A ranking system based on multi-objectives is used to determine the most suitable trained agent for testing. The system is based on the number of halting vehicles, the ratio of clearance, the mean waiting time, the mean travel time, and the mean cruising speed. This measuring technique is practical because computational power has been exceeded. For further details on trained agent selection, refer to Appendix D.

4.5 Testing and Evaluation

The testing is an online stage where the deployed controller monitors and manages the junction using the trained memory and neural architecture. The testing of the developed controllers is carried out on a different set of traffic data. This arrangement is necessary to ensure that the agent during the learning process is not overfitting and is capable of performing in alternative scenarios. Besides that, having a separate dataset will not give the developed intelligent controllers an advantage over the other comparative systems.

Table 4.14: Traffic volume for testing sets

Model	Isolated	Network
Data Set	8:00-9.00am	
Traffic Volume (veh/hr)	5,439	15,508[B]

[B]Reported volume after balancing the network of 45,563veh/hr

4.5.1 Comparative Systems

The comparative controller programmes include all generations of traffic controllers: fixed, actuated, and adaptive. The fixed controller represents the site condition of the traffic data. The time plan and cycle assignment correspond to the site condition during data collection.

4.5.1.1 Fixed Controller

The fixed controller was driven from Webster's theory. The objectives of Webster's technique include (i) the development of shorter queues for

traffic streams, (ii) the minimization of total vehicle delays, and (iii) the increment of the intersection's throughput (Krishna et al., 2018). The optimum cycle length C_{opt} is the ratio of total lost time L to total critical flow ratio Y. This ratio is presented by equation 4.17 (Zakariya and Rabia, 2016).

$$C_{opt} = \frac{1.5L+5}{1-Y} \tag{4.17}$$

4.5.1.2 Delay-based Actuated Controller

The actuated signal controller based on delay time (Delay) approach is an actuated generation strategy proposed by Oertel and Wagner (2011). The system adjusts the green duration by utilising vehicles' delay. The green phase is terminated as soon as the accumulated delay on an approach is dissolved. A single delay d_i occurs within a time increment Δt when the current speed of a vehicle v_i cannot reach its maximum speed limit v_{max}. The summation of d_i gives the delay in an approach d as in equation 4.18. The Delay system was initially developed for isolated intersections and was reported to outperform traditional strategies with penetration rates above 10%.

$$d = \sum_{i=1}^{n} \Delta t \left(1 - \frac{v_i(t)}{v_{max}}\right) \tag{4.18}$$

4.5.1.3 Longest-Queue-First Controller

The longest-queue-first algorithm (LQFA) is online adaptive control logic. The LQFA aims to minimise queue size in each direction of traffic flow for a junction (Wunderlich, 2007). This queue-based scheme prioritises lanes with larger queue lengths (Wu et al., 2017). The original Wunderlich's algorithm has a weighting factor for a certain class of vehicles (e.g., emergency). However, considering the vehicles carry the same weight in this study, the algorithm only focuses on maximising queue output per phase. The signal phase $\vec{p}$ corresponds to the maximum number of vehicles queued at a current time Q_t (queue occupancy vector), as in equation 4.19.

$$\vec{p} \in \max(Q_t) \tag{4.19}$$

4.5.1.4 Maximum Pressure Controller

The online adaptive maximum pressure control algorithm (MaxPressure) responds to the maximum product of weighted queue length γ and its corresponding saturation flow *S* for each phase (Varaiya, 2013). The γ is the difference between the upstream and downstream queue lengths. The greedy policy aims to increase throughput at the signalised junction. The MaxPressure *u* at every state *X* is given as in equation 4.20.

$$u(X) = argmax\{\gamma(S)(X)\} \tag{4.20}$$

4.5.1.5 Actor-Critic Reinforcement Learning Controller

The AC-RL is an RL algorithm enclosing an actor that selects actions and a critic that gets the agent near to long-term objectives (Aslani et al., 2017).

The above systems (excluding Fixed) are developed based on a similar acyclic programme for this research's proposed intelligent agents. This design consideration is vital to ensure that system variations are technically associated with traffic policy and to rule out any bias that could arise from inequivalent timing plan strategies during the testing and evaluation stage.

4.5.2 Traffic Environment Models

We developed two (2) DRL control logics (DCNN and DQLA k-v) to achieve the objectives and close the current gaps in the DRL studies. Two (2) stochastic traffic micro-model environments (isolated and arterial) were calibrated and validated using real-traffic conditions.

Each intelligent controller has a different traffic control strategy. DCNN executes an upstream strategy (waiting time), a popular technique in present DRL studies. DQLA *k-v* is based on a downstream capacity policy. The novel proposed downstream policy directly addresses traffic flow and intersection capacity.

Both DRL controllers (DCNN and DQLA *k*-*v*) have similar deep learning structures, and their performance variation is anticipated to be directly related to the reward control policy. Table 4.15 summarises each testing model environment and the comparative systems used.

Table 4.15: Testing Models and comparative systems

Test Model	Agent/ Policy	State Representation	Signal Reward	Comparative Systems	Testing Objectives
Isolated Intersection	DCNN/ Upstream	Speed, Position and Traffic State	Waiting time	• Fixed • LQFA • Actor-critic RL	• Efficiency of the DRL agent in a stochastic traffic model of the environment • Stability of DRL in various traffic conditions • Implementing built-in infrastructure detection technology for the DRL system
Network	DQLA *k-v*/ Downstream	Density	Speed and density at exit links	• Fixed • Actuated-delay • LQFA • MaxPressure	• Scalability of a single system for network operation • Sustenance of control strategy • Compatibility of communication channel for optimising signal operation in the arterial network • Evaluating the system in a realistic and stochastic traffic model of the environment

4.6 Summary of Methodology

The deep learning technique for the logic design is intended to advance the signal operation by predicting near-future changes in the intersection environment. To ensure that the controller has the ability to meet the demand, the acyclic plan is integrated into the controller system. Thereof, an executed phase will serve the demand instantaneously. The phase duration meets the standards and requirements for safe operation.

To scale up the design of the controller and suitability for present integration using available detection technology, both developed controllers rely on data input from detector devices within a defined area of the intersection. The detection zone is within a practical range of not more than 140 metres from the stop line. Furthermore, both of the developed controller logics are decentralised and do not require coordination with neighbouring junctions. The single design is preferred in order to reduce the complexity and system requirements that are often associated with centralised and coordinated systems. The centralization and coordination implicate the applicability of DRL deployment on a real-world scale.

To deal with the issue of evaluation and validate the performance of the DRL controllers, an accurate model of the study area was developed. The features of the model comprise five (5) vehicle classes, various junction configurations, road geometric elements, and behavioural attributes to accurately represent patterns and driving conditions in the environment. In

total, two (2) models were developed for an actual study area in Malaysia, including (i) the isolated intersection and (ii) the arterial network. Each of these evaluative models is required to validate a number of objectives.

The formulated procedure in this research work is comprehensive and takes into account aspects of system features, evaluation, and operation. This methodological approach is believed to be necessary to answer the research questions and is meant to close the mentioned gaps (scalability, sustenance, and evaluation) that were cited in this thesis work. Figure 4.13 presents the procedure summary.

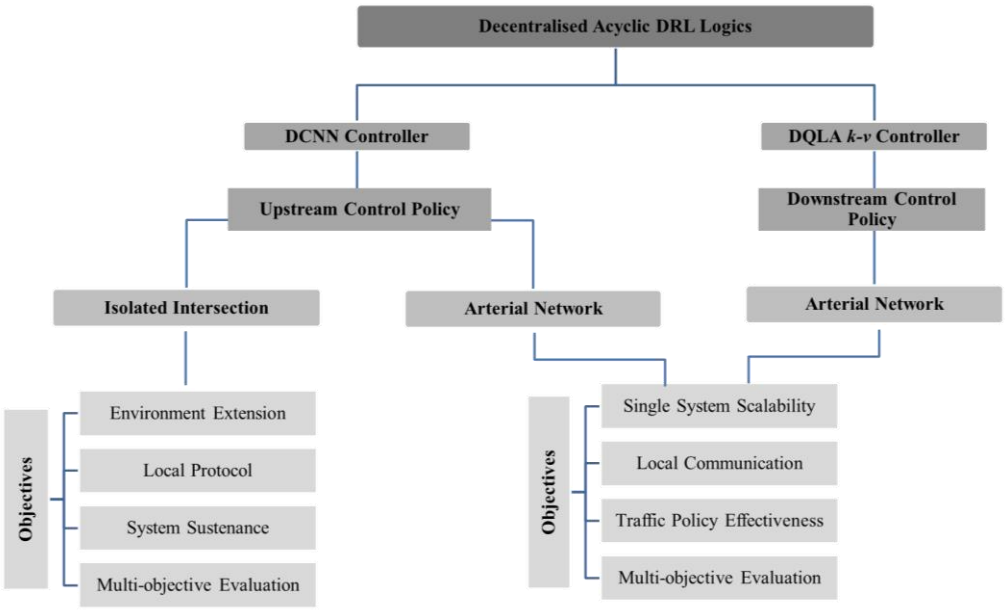


Figure 4.13: Summary of the procedure

CHAPTER 5

RESULTS AND DISCUSSIONS

The chapter is segmented into two (2) sections based on test bed type. Section 51 presents the findings related to the isolated traffic signal environment, and Section 5.2 presents the findings for the network model environment. All the reported results take into account the midnight effect of the simulation. The early minutes of the simulation run are disregarded from the analyses. In addition, the average measured attributes are based on a number of iterations corresponding to the 95% confidence level. This repetition of model runs is significant to ensure that the results are valid and reflect real-world conditions.

5.1 Isolated Signal Operation

Four (4) model scenarios corresponding to various levels of traffic saturation flow were developed to evaluate the controller's stability and the proposed training method. The quantitative analysis is essential to benchmark the DCNN agent performance under various traffic conditions and to weigh the DCNN agent’s stability. The traffic scenarios are as follows:

- Low Saturation Environment (L-Sat Env.): the capacity utilisation is below 36%. Low arrival rate to the intersection at 3,284veh/hr.

- Medium Saturation Environment (M-Sat Env.): the moderate utilisation of the capacity between 36% and 66%.
- High Saturation Environment (H-Sat Env.): this scenario represents the actual surveyed junction condition with traffic flow amounting to 5,439veh/hr. The capacity utilization is 83%.
- Over Saturation Environment (Over-Sat Env.): the total traffic flow at the junction exceeds capacity (>100%). This environment represents severe traffic conditions with high flow rates (6,984veh/hr). Signal controllers typically fail to adapt and mitigate such traffic events.

Each saturation condition was tested for the following systems: (i) fixed (FC), (ii) longest-queue-first (LQFA), (iii) actor-critic RL (AC-RL), and (vi) the proposed deep convolution neural network (DCNN) algorithm.

Three (3) aspects to examine the impact of the signal systems influence on (i) traffic attributes in terms of time and speed, (ii) traffic flow and simulation run time, and (iii) phase time and traffic demand. In addition, the proposed DCNN was benchmarked against other prominent DRL systems.

5.1.1 Timing and Speed Performance Measures

The t-statistic test was conducted to evaluate the significance of the reported difference between DCNN and comparative signal systems. Table 5.1 presents the test findings, followed by a detailed discussion.

Table 5.1: Measure of performance for various traffic attributes

Test Set	L-Sat Env			M-Sat Env.			H-Sat Env.			Over-Sat Env.			μ. Under-Sat
System	μ.TT (s)		p-value	μ.TT (s)		p-value	μ.TT (s)		p-value	μ.TT (s)		p-value	μ.TT (s)
	DCNN = 74.32	Diff. (%)		DCNN = 93.30	Diff. (%)		DCNN = 101.16	Diff. (%)		DCNN = 101.80	Diff. (%)		
FC	90.03	-17%	<0.05	106.26	-12%	<0.05	135.23	-25%	<0.05	163.64	-38%	<0.05	-18%
A-C RL	71.43	4%	0.62	92.3	1%	0.81	97.98	3%	0.35	109.32	-7%	<0.05	
LQFA	63.27	17%	0.09	79.94	17%	0.07	94.16	7%	0.06	98.73	3%	0.09	
System	μ.WT (s)		p-value	μ.WT (s)		p-value	μ.WT (s)		p-value	μ.WT (s)		p-value	μ.WT (s)
	DCNN = 0.33	Diff. (%)		DCNN = 0.28	Diff. (%)		DCNN = 0.91	Diff. (%)		DCNN = 9.05	Diff. (%)		
FC	0.33	0%	0.47	0.57	-51%	0.13	40.87	-98%	<0.05	184.6	-95%	<0.05	-91%
A-C RL	0.37	11%	0.36	0.29	-3%	0.66	0.24	279%	0.32	10.91	-17%	0.71	
LQFA	0.33	0%	0.44	0.81	-65%	0.24	5.6	-84%	<0.05	61.51	-85%	<0.05	
System	μ.RS (m/s)		p-value	μ.RS (m/s)		p-value	μ.RS (m/s)		p-value	μ.RS (m/s)		p-value	μ.RS (m/s)
	DCNN = 0.39	Diff. (%)		DCNN = 0.28	Diff. (%)		DCNN = 0.21	Diff. (%)		DCNN = 0.14	Diff. (%)		
FC	0.31	26%	<0.05	0.24	17%	<0.05	0.16	31%	<0.05	0.09	56%	0.05	2%
A-C RL	0.41	-5%	0.38	0.29	-3%	0.59	0.22	-5%	0.07	0.16	-13%	0.05	
LQFA	0.51	-24%	<0.05	0.37	-24%	<0.05	0.25	-16%	<0.05	0.24	-42%	0.05	

μ.WT (s) = mean Waiting Time, μ.TT = mean Travel Time, μ.RS (m/s) = mean Relative Speed (m/s), Diff = Difference between DCNN to a comparative logic

μ-Under-Sat: mean performance computed for under-saturated conditions (low, medium and high test scenarios) where $p<0.05$

DCNN vs. FC: the measure of performance attributes indicated that the proposed DCNN surpassed the site condition in various test scenarios. The recorded values showed travel time savings between 12% and 38%. The saving gap increases almost linearly with the scenario challenge. Between L-Sat and Over-Sat, the mean travel time under the FC system rose from 90 seconds to 164 seconds, or an 82% increment. In comparison, DCNN showed an increment of only 37% (from 74 seconds to 102 seconds). On the same basis, the mean waiting time experienced by vehicles increased to three (3) minutes for FC in Over-Sat conditions compared to less than 10 seconds for DCNN. The significant improvement in operational conditions for DCNN is related to two (2) factors: (i) the real-time adjustment to traffic demands and (ii) the controller's policy associated with halting time at the intersection level. In contrast, FC is rigid as signal timing is predetermined based on assuming constant traffic flow.

DCNN vs. A-C RL: these memory-based RL systems were designed using similar reward functions (i.e., waiting time). Hence, it is not surprising to find out that their performance is indistinguishable in terms of waiting time ($p<0.05$). The only cited significance is in travel time for the Over-Sat scenario. In the Over-Sat model, DCCN significantly reduced travel time by 7%. The relative speed gap produced the time gains. The relative speed between moving and halting vehicles was significantly reduced by 13% under the DCNN controller. Putting into account that the training conditions of both algorithms were similar, the finding indicated that the policy-driven RL

algorithm was possibly better at managing operations in a new environment than the temporal difference RL algorithm (A-C RL).

DCNN vs. LQFA: The superiority of DCNN is evident in the H-Sat and Over-Sat test beds. Under these traffic flow conditions, DCNN produced lower waiting times (reward policy), closing the gap between halting and moving vehicles. There is a flux in waiting time by 10 folds between the H-Sat (6 seconds) and Over-Sat (62 seconds) under LQFA operation. On the other hand, LQFA barely increased by 4 seconds (4%) in travel time and differed by 0.01 (4%) in mean relative speed between the H-Sat and Over-Sat conditions. LQFA optimises operations based on queue length. Hence, as the traffic arrival rate fasted and became almost equal in all directions of the intersection (filling detection space), LQFA converged to vehicle platoon optimisation. The finding presents the importance of detection boundary limits for developing controllers.

The log travel time for all test scenarios showed that RL-based controllers achieved moderate travel time during the peak hour simulation. Figures 5.1 to 5.4 present the log travel time.

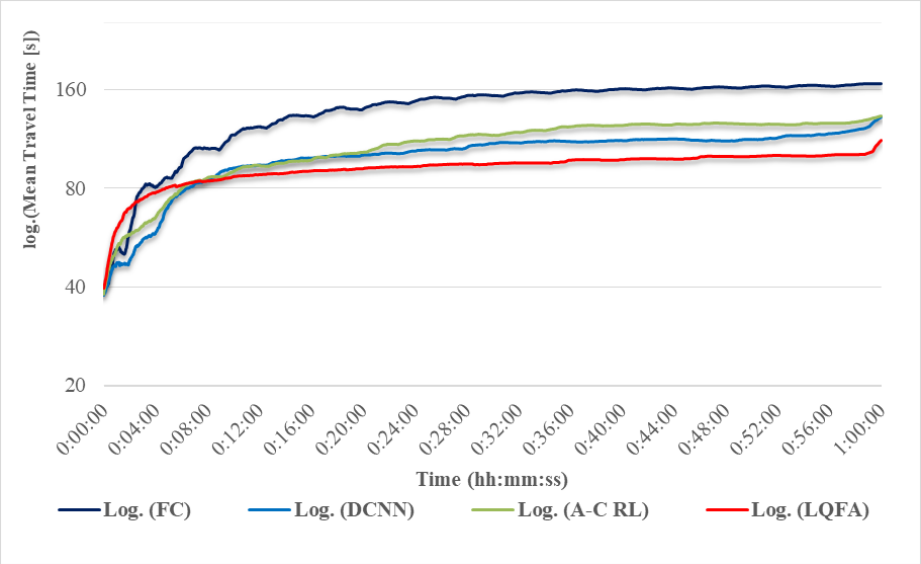


Figure 5.1: Log travel time for over-saturated (Over-Sat) scenario

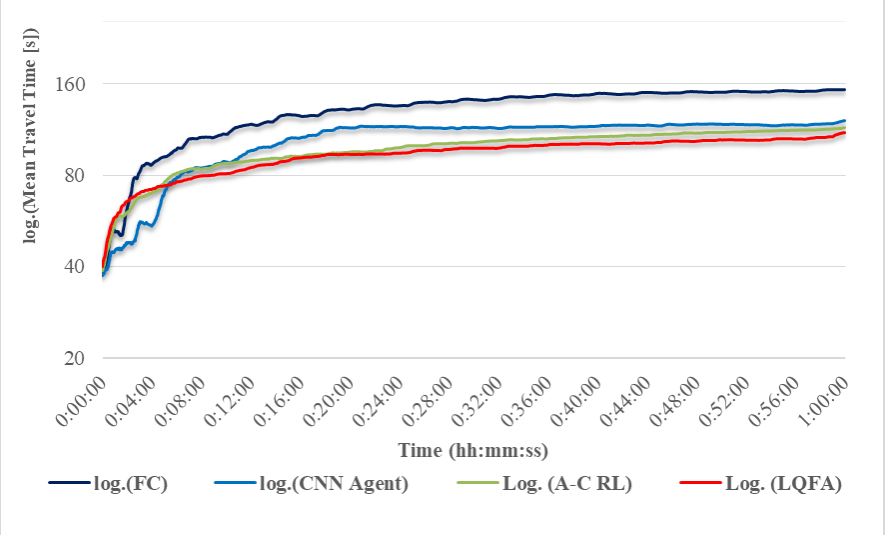


Figure 5.2: Log travel time for high saturated (H-Sat) scenario

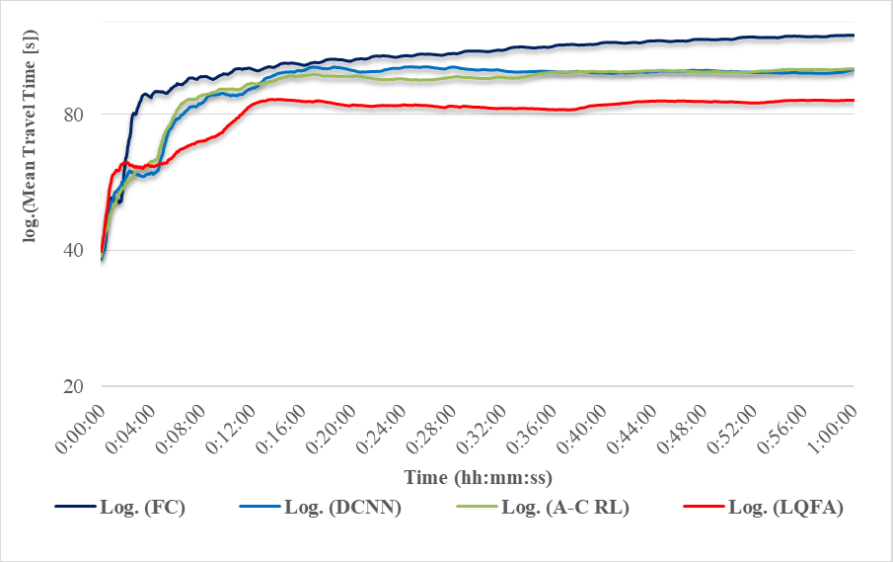


Figure 5.3: Log travel time for medium saturated (M-Sat) scenario

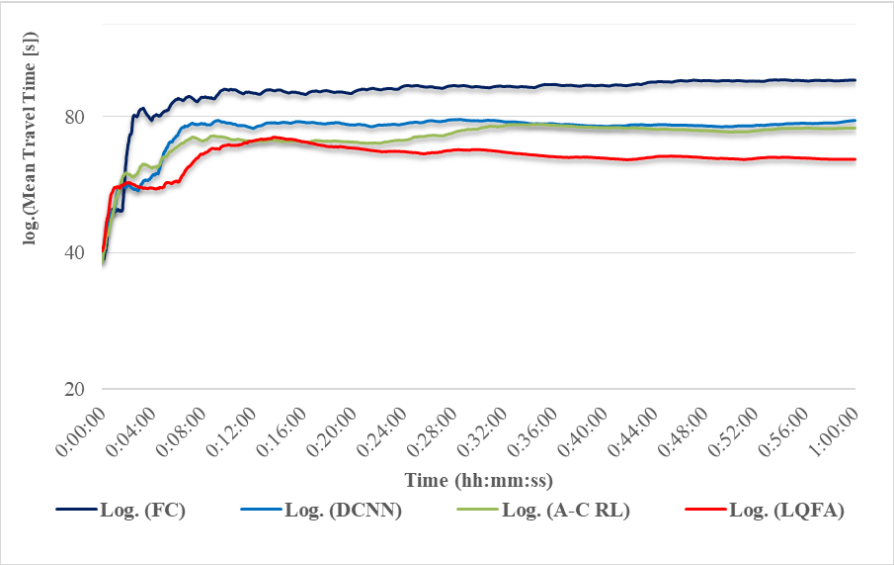


Figure 5.4: Log travel time for low saturated (L-Sat) scenario

5.1.2 Flow Rate and Simulation Run

The traffic volume is peak hour counts at the junction border. All flow rates correspond to 3,440 seconds, which was found suitable during the

calibration and validation process. As all lanes lead to the signalised intersection, the throughput will directly impact the time of the simulation run (completion). The simulation run is the total time duration needed for each vehicle to complete its trip and exit the model. Alternatively, the simulation end time will take longer to terminate whenever the vehicle flow is disrupted. The sources of disruption are related to (i) delay at the junction level due to stopping and (ii) delayed insertion into the model if there is a backlog from the junction (shortage of space). The active vehicles are defined as the count of running vehicles per unit time (veh/s). For simplicity, the term active vehicles is interchangeable with flow rate (veh/s). This performance measure is significant in determining the effectiveness of the signal logic in mitigating throughput. The whisker analysis is utilised for this section.

Over-Sat Env.: Based on iterative runs for oversaturated flow conditions, the proposed DCNN system acquired the most optimal performance and led to the highest median flow rate at 295veh/sec, and the least simulation time at 63 minutes. The FC system recorded a close median flow with nearly a 1% difference (292veh/sec) to DCNN, but FC had twice the simulation time (133 minutes) to reach the mentioned flow record. In other words, in a real-world situation, the pre-timed programme will need more than two (2) hours to clear the imbalanced traffic condition. This time duration for FC logic is twice the needed time for DCNN logic to deal with similar traffic conditions.

Though LQFA showed the second slowest signal operation (after FC) at 83 minutes for averaged simulation runs. The recorded time of LQFA is 30% more than the proposed DCNN to clear the rush hour traffic volume. In addition, LQFA registered the lowest mean flow rate at 202veh/sec, or a 29% lower flow value compared to DCNN. From these findings, it seems that the policy based on queue length diminishes in imbalanced conditions and when the difference among traffic volumes in competing directions reduces. This suggests that LQFA works best when there is an apparent hierarchy in traffic demands.

Comparing the flow rate and simulation run for DCNN and A-C RL, the former showed a 5% higher average flow rate and a 3% shorter simulation run. Though these results might not be of much significance in the isolated intersection context, in a large-scale network, these performance achievements are translated to greater reward and mitigation gains. The following Figure 5.5 presents the flow rate and simulation duration for the Over-Sat model.

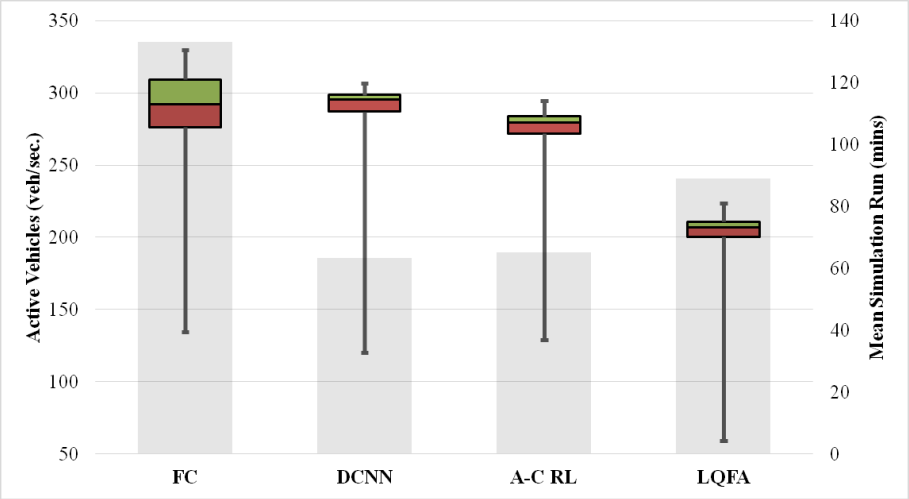


Figure 5.5: Flow rate (primary access-left) and simulation time (secondary access-right) for Over-Sat environment

H-Sat Env.: The whisker analysis provided information on the flow rate distribution (veh/s). Results indicated that FC had the highest flow rate at 220veh/sec. The pre-timed controller was designed to correspond to the saturation flow of the site condition. However, the FC system's rigid time phasing does not change in real-time, causing the fixed timer control to take longer to mitigate the operation at the intersection level. The proposed DCNN system effectively serves demand, and the traffic input is cleared 15 minutes faster (or 19% improvement) than the FC system.

Other comparative systems (LQFA, and A-C RL) showed comparable performance in clearing the model's traffic with nearly a 2% (1~2minutes) time difference compared to the proposed DCNN system. Nonetheless, the mean traffic flow for the proposed DCNN controller was improved by 7% and 21% compared to A-C RL and LQFA, respectively. The improved flow aligns with the earlier gains in timing and speed attributes. The following Figure 5.6 presents the flow rate and simulation duration for the H-Sat environment.

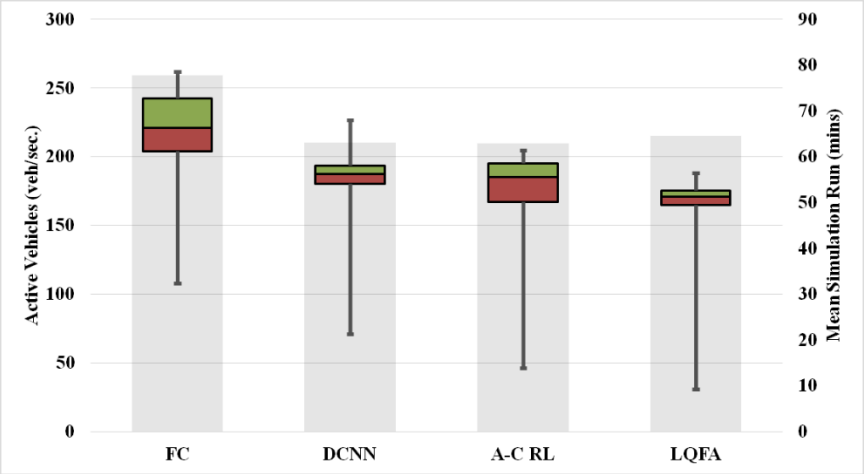


Figure 5.6: Flow rate (primary access-left) and simulation time (secondary access-right) for H-Sat environment

Med-Sat and **L-Sat Env.:** In under-saturated scenarios, the FC system maintained the highest median traffic movement rate compared to other controllers. Simultaneously, the pre-time logic produced the highest spread in flow ratio. The wide range of flow is caused by the traffic policy, which corresponds to saturation flow. Longer phase time is given to directions with higher demand, and vice versa.

In comparison, the proposed DCNN controller minimised flow spread distribution (fairer policy) for both test conditions and cleared directional flow within approximately 2% of the other controllers. The results indicate that DCNN is superior in responding to traffic demand and treating directional flow faster and fairer. In the L-Sat, DCNN lowered the distribution between 3% and 38%. In the M-Sat, DCNN improved the distribution by 12% to 34%. Figures 5.7 and 5.8 present the flow rate and simulation duration for the M-Sat and L-Sat environments.

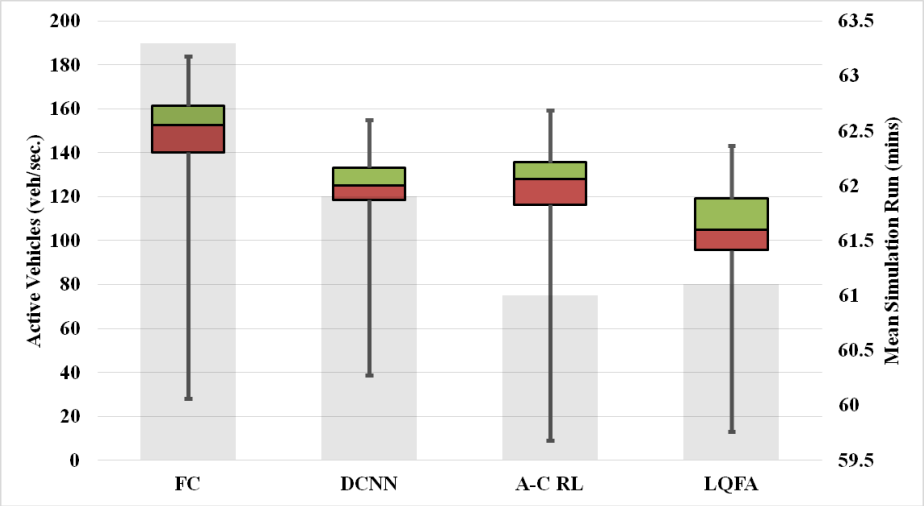


Figure 5.7: Flow rate (primary access-left) and simulation time (secondary access-right) for M-Sat environment

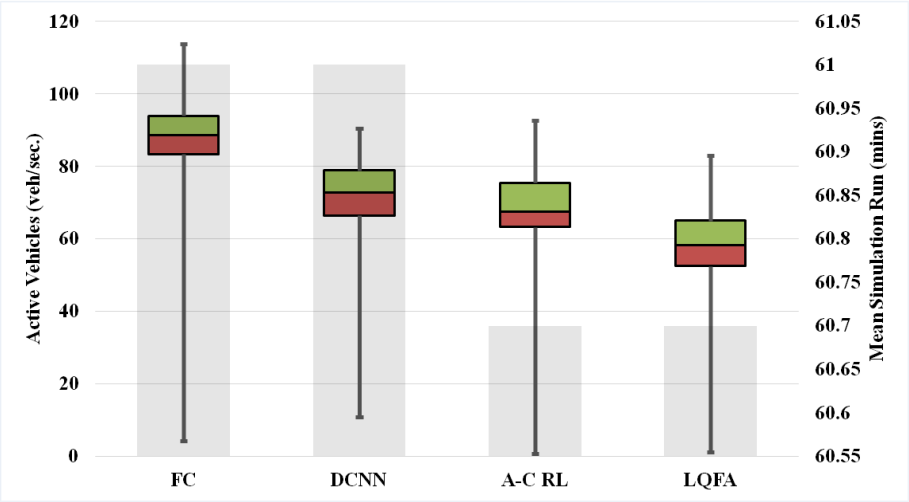


Figure 5.8: Flow rate (primary access-left) and simulation time (secondary access-right) for L-Sat environment

The flow rate and run time analyses verify that the proposed DCNN achieved the best performance, especially in imbalanced saturation conditions. The over-saturated test verified that the conventional system (FC) and online controller (LQFA) had difficulty mitigating signal operations. The traffic policy for both systems causes the deficiency. The classic fixed controller considers fixed arrival rates, whereas LQFA requires slow changes in traffic

dynamics. Evaluating the difference in run time between under-saturated and over-saturated scenarios, the efficiency in signal operation dropped by 49% and 30% for FC and LQFA, respectively. In actual traffic applications, these deficient values mean that signal lights will require additional time between 20 and 30 minutes to clear the one-hour traffic flow.

The memory-based controllers showed a small deviation in signal operation between 2% and 6% for the proposed DCNN and A-C RL controllers, respectively. On this ground, the value-based logic (DCNN) surpassed the actor-critic (A-C RL). The finding indicates that the proposed DCNN signal logic and policy were trained and implemented successfully, contributing to its stable performance. Table 5.2 presents a summary of the flow rate values and simulation runs.

Table 5.2: Summary of Whisker analyses and data in terms of statistics

Scenario		Over-Sat Env				H-Sat Env				M-Sat Env				L-Sat Env				Under-Sat Env			
Controller		FC	DCNN	A-C RL	LQFA	FC	DCNN	A-C RL	LQFA	FC	DCNN	A-C RL	LQFA	FC	DCNN	A-C RL	LQFA	FC	DCNN	A-C RL	LQFA
Whisker Box Values for Flow Rate (veh/sec)	Minimum	134	120	129	59	108	71	46	31	28	39	9	13	4	11	1	1	47	40	19	15
	Q1	276	287	272	200	204	180	167	165	140	119	116	96	83	66	63	52	143	122	116	104
	Median	292	295	280	207	221	188	185	171	153	125	128	105	89	73	68	58	154	128	127	111
	Q3	309	299	284	211	243	193	195	175	161	133	136	119	94	79	76	65	166	135	135	120
	Maximum	330	307	295	224	262	227	205	188	184	155	159	143	114	90	93	83	187	157	152	138
	Range	195	186	166	165	154	156	159	157	156	117	151	130	110	80	92	82	140	117	134	123
Data	Average Flow Rate (veh/sec)	287	283	268	202	218	185	177	163	148	125	124	106	87	71	68	58	151	127	123	109
	Average Simulation Time (min.)	133	63	65	89	78	63	62	64	63	62	61	61	61	61	61	61	67	62	62	62

5.1.3 Signal Phasing Controller and Traffic Demand

Cycle-free means choosing the appropriate phase assignment without following a particular order of green time allocation or road hierarchy. The acyclic plan was embedded in three (3) logics (DCNN A-C RL and LQFA). The FC controller presented the site condition. A major player in assigning the phase signal depends on the travel demand at the intersection.

The approach's demand for the test set showed that the north approach had a higher traffic volume at 2,005 or 37%, followed by the southern approach at 33% (1,815 vehicles), the eastern approach at 24% (1,316 vehicles), and lastly, the west approach at 6% (with 303 vehicles).

Generally, the traffic phasing corresponded proportionally to the approaches with the highest demand volume (northern and southern directions). This correspondence is true for the acyclic-based controllers, as shown in Figure 5.9.

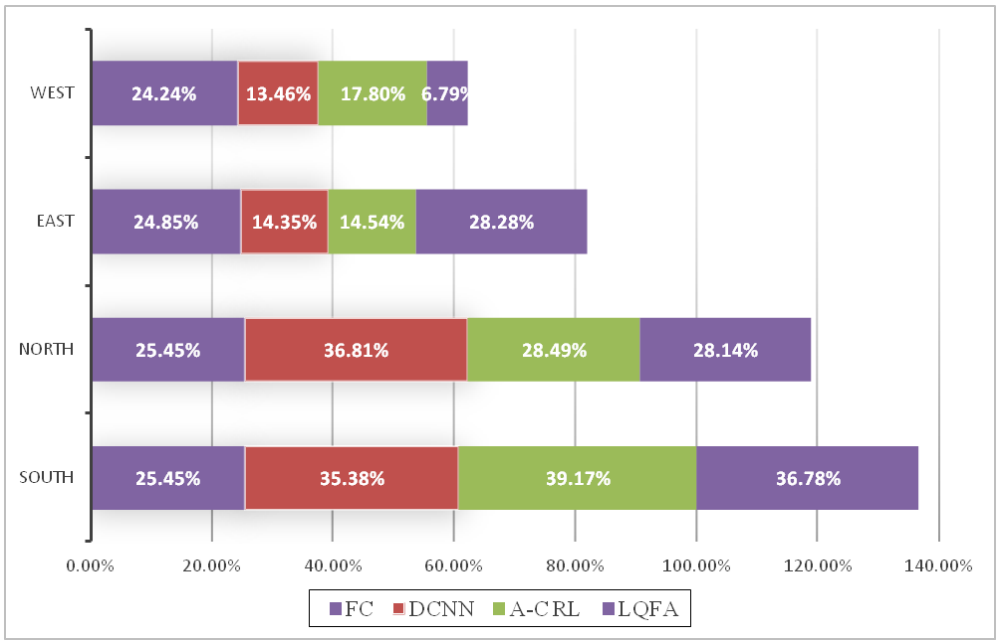


Figure 5.9: Proportion of traffic light phase per controller

On the other hand, lower travel corridors had varied findings. Where the RL-based controllers intended to give lower phase allocations, the queue-based controller (LQFA) gave a higher allocation to the east leg of the intersection. Breaking down the volume at this intersection showed that the traffic volume at the east approach includes 59% (uncontrolled) left-turning movement in the test set. The traffic signal does not control the short storage left-turn lane. The memoryless online controller (LQFA) likely corresponded to the spillback queue and eventually allocated more phase timing for the eastern approach. The mitigation had impacted the signalised junction's global optimisation, leading to higher time costs and lower throughput, as in Sections 5.1.1 and 5.1.2.

In comparison, the RL-based controllers (DCNN and A-C RL) were much more capable of anticipating traffic dynamics using the taught memory

to configure the phasing assignment. The operation assignment depends on the forecasted input (vehicle speed, position, and signal phase) and expected gains (minimising the halting time between the red and green directions). Based on the results, the green movement allocation at major approaches with higher traffic demand increased by 42% for the peak hour test. In contrast, the west and east approaches witnessed a simultaneous reduction of 43% and 33% for each dataset, respectively. In addition, the results prove the importance of acyclic design to boost the performance of the RL-based controllers. Furthermore, the ability of RL-based controllers to evolve to optimal solutions in stochastic environments with proper training and a bias-free guidance policy.

5.1.4 Benchmarking to DRL Studies

The purpose of this section is to evaluate the performance of the proposed DCNN (simple structure) with other similar and complex DRL structures from the literature review chapter (Section 2.7). It is worth noting that the DRL studies had different experimental settings and traffic flow aspects, making this benchmarking qualitative rather than quantitative. The qualitative approach determines the extent of the design aspect in the agent's performance.

The experimental context in this PhD study is more complex compared to other DRL studies for the following reasons:

1. Stochastic traffic environment. The traffic flow is not uniform. The variation in traffic flow behaviour reaches 20%. The micro-model does not have a fixed seed value.
2. Mixed-mode traffic model. The model comprises five (5) classes of vehicles. Each class exhibited unique behaviour and was calibrated and validated individually against site conditions.
3. Illustrative case study. The intersection layout and volume represent site condition. The traffic volume stands at 5,439 vehicles (H-sat environment). The intersection layout has uncontrolled turning movements. The approach lanes differ in the number of lanes. All turning movements are considered for the signal operation.
4. Training and testing environment. Unlike many DRL studies, the DCNN controller was trained and tested on a different traffic data set. Training and testing on the same traffic data set is biased as it gives the memory-based controller an advantage in knowing its test model and better chances to exceed other comparative controllers.
5. Various testing scenarios based on intersection capacity were developed in this study. The traffic volume for DCNN test scenarios ranged from 3,284 vehicles (L-Sat) to 6,984 vehicles (Over-Sat). This study's lowest traffic flow ratio in the L-Sat scenario is comparable to other DRL studies' commonly reported flow ratios for the isolated signal intersection.

These five (5) environmental characteristics make the testing bed much more challenging for the proposed controller. Comparatively, the DRL studies integrated some (but not all) of the settings mentioned above.

The DCNN architecture is similar to popular studies as in Genders and Razavi (2016) and Gao et al. (2017), and DCNN is less complex compared to DDPG (Casas, 2017), 2DSARSA (Yen et al., 2020), and 3DQN (Liang et al., 2017). The state representation is DTSE, with adjustments to accommodate vehicle categories. The action of DCNN is acyclic. In this aspect, less than 30% of studies proposed this signal assignment and mainly focused on restricting the role of the adaptive RL controller to phase execution. While the authors intended to propose short durations (<5 seconds), the developed DCNN logic executed an effective green time of 20 seconds. The choice of long duration is to reduce the causes of variation in experimental settings to evaluate the DRL agent's effectiveness in a stochastic environment against real signal operation settings (pre-timed controller).

Overall, the DCNN controller has more similarities in control agent design and state representation but differs in environment settings and action plan from other DRL controllers.

Two (2) popular traffic measures in traffic control studies are often reported: travel time and waiting time. Based on Figure 5.9, the proposed DCNN agent achieved the most waiting time improvement at 90%, and it marked about a 10% improvement over the nearest controller agent.

Regarding travel time, the proposed controller yielded the highest saving in travel cost (20% improvement) and performed equivalently to the control agent presented in Genders and Razavi (2016). These findings indicated that the proposed DCNN logic converged better in the operating signal environment.

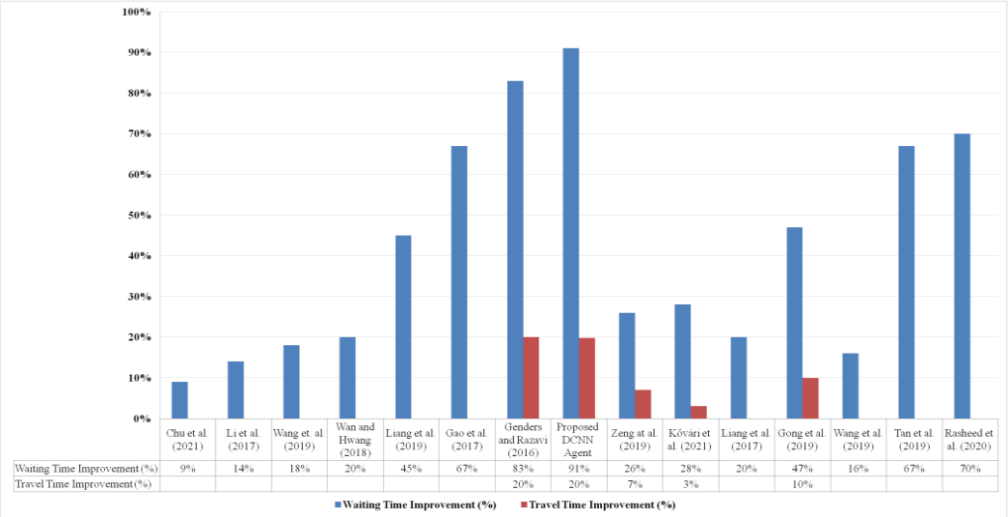


Figure 5.10: Benchmarking the mean performance of DCNN and other DRL controllers from the literature review

The other main observations are as follows:

Traffic policy: the control strategy for DCNN is commonly used (57%) in DRL studies. The architecture of DCNN design is comparable to Genders and Razavi (2016), Gao et al. (2017), Kővári et al. (2021), Wan and Hwang (2018), and Chu et al. (2021), and DCNN is less complex compared to DDPG (Casas, 2017), 2DSARSA (Yen et al., 2020), and 3DQN (Liang et al., 2019, and Wang et al., 2019). The DTSE of DCNN is often utilised for state representation in nearly 57% of the reported DRL studies. The reward policy

is not uncommon and is addressed in several DRL techniques. Of further interest, the traffic volume used to test the proposed controller is high compared to other studies. Nonetheless, the proposed DCNN logic showed the highest gains. The comparative discussion indicates that the method of training DCNN successfully yields better results. Hence, the extension of accurate representation of the environment model is evident to impact the intelligent controller performance.

5.1.5 Closing Remarks for DCNN Controller

Gao et al. (2017) defined stability in control decision when no oscillation between good and bad action is observed. Several DRL studies showed that their proposed agents were unstable enough to maintain stability in various testing conditions. The studies of Casas (2017), Li et al. (2016) and Chu et al. (2021) reported deficiency challenges in various testing conditions. In contrast, the DCNN system could maintain stable performance in the alternative, non-trained environments.

Another issue is the communication protocol, where almost two-thirds of the studies considered the environment fully known to the controller. This assumption led to exponentially growing state representation and limited the practicality of DRL controllers for large networks. It is essential to include the multimodal traffic environment in the control theory (Wang et al., 2018).

To address these gaps, we focused mainly on the environmental aspect, where little attention was given to this part of the study. Since the RL agent learns in a model-free context, it is necessary that the environment model mimics the actual traffic conditions and reflects accurate changes in relation to the agent's decision (Han, 2018). This is the first study to consider a heterogeneous micro-model to train and test the DCNN algorithm. The training environment needs to be stabilised to train an effective DRL algorithm. To test this hypothesis, we integrated a real isolated signalised junction.

The micro-model has heterogeneous features and was calibrated and validated accurately. The state-space definition was based on the adjusted DTSE to suit the mixed road and user classes. The data feed is bound to 70 metres of road length. The signal plan is acyclic with a minimum phase duration of 16 seconds to keep the system close to site condition (fixe system). The memory-based agent was trained to 95% CL to ensure effective learning. The 95% CL is a popular threshold in traffic engineering modelling assignments.

The examined results verified that the proposed method stabilised the DCNN agent. Though training and testing differed in traffic volume, the memory-based controller converged to optimal operation and outperformed other comparative systems in over-saturated and under-saturated operational conditions. The benchmarking also showed the superiority of DCNN over other "similar" and "complex" DQN agents' structures. The present

accomplishments verify the importance of the environment in stabilising intelligent controllers. The following findings are summarised as follows:

1. Extend of the environment to stabilise the operation of the intelligent controller. DQN training needs to be integrated with the appropriate representation of the model environment.
2. Local feed channel is sufficient for deep signal control. Built-in infrastructure detection tools can be incorporated with intelligent adaptive controllers. The fully observed and unbounded assumption of the environment is not necessary for DQN controllers.
3. Unlike other problems, the traffic dynamics can be generalised to address control operations. The agents need to learn suitable control policies to optimise signal timing.
4. Appropriate traffic signal phasing. Using a short-phase signal gives the advantage to any controller. Therefore, research studies must observe caution when determining signal phasing and validate DRL controllers using proper and fair time plans for test settings.
5. These extra granted seconds to an approach could lead to longer waiting times and worsen the operational performance. Therefore, it is vital to liberate the signal system towards a more practical green time phasing where minimum phase allocation is granted to ensure the safety and practicality of the

signalised junction and fair distribution based on traffic volume demand and lane occupancy.

5.2 Arterial Network Operation

The efficiency of the DRL adaptive signal controller is particularly challenging in the context of urban network operation. This study introduces a control strategy based on intersection capacity (DQN *k*-*v*). The optimisation technique is formulated based on the available space at the discharge zone. This analysis aims to test the efficiency of DQLA *k*-*v* logic versus other signal controllers, including the earlier proposed DCNN logic. Based on Section 5.1, DCNN surpassed comparative system controllers. In this section, DCNN action is adjusted to a similar control plan as DQLA *k*-*v*. This control plan adjustment eliminates any bias that could arise from a long signal duration. The micro-model of the urban network environment consists of nine (9) signalised junctions.

The statistical analyses are segmented into four (4) categories, including (i) time and speed factors, (ii) traffic flow clearance ratio, (iii) number of experienced stops, and (iv) network-wide time loss. The analyses are based on mean values based on several iterations to capture traffic dynamics.

5.2.1 Waiting Time, Travel Time and Travel Speed

The hypothesis test statistic (*t*-statistics) was utilised to verify the significance of DQLA *k-v* over other comparative logics.

DQLA *k-v* vs. Fixed: the proposed DQLA *k-v* control method showed superior performance ($p<0.05$) in terms of mean waiting time and travel time compared to the Fixed technique. Vehicles traversing the network experienced 10% savings in halting time and 5% shorter travel time. On the other hand, both controllers showed a similar mean travel speed of 3.40m/sec.

DQLA *k-v* vs. Delay: based on recorded data, the proposed method optimised cursing speed by almost three (3) folds compared to the actuated controller. As vehicles traversed at higher speeds, they experienced a lower travel time of 12% across the 7.5km network. Despite DQLA *k-v* leading to a shorter mean waiting time of 2.3%, this result was insignificant at $p>0.05$. The policy strategy of the Delay controller is to reduce the delay of competing traffic demands and terminate the green phase once the accumulated delay on an approach is dissolved. Delay's strategy delivered its purpose but was not flexible enough to accommodate other traffic measures.

DQLA *k-v* vs. LQFA: The online optimisation using LQFA performed the worst in terms of mean timing parameters. LQFA strategizes queue length to manage the operation. In this aspect, low-traffic demand approaches were held for longer, leading to higher accumulative costs at the

network operation level. On the other hand, the *k-v* policy had a balanced approach to equating signal chances based on available capacity downstream, leading to better signal plans and reducing waiting and travel time costs by about 37% and 23%, respectively. In addition, DQLA significantly ($p<0.05$) improved travel speed by 29% compared to LQFA.

DQLA *k-v* vs. MaxPressure: mean timing elements did not significantly differ between DQLA *k-v* and MaxPressure. On the other hand, DQLA *k-v* surpassed ($p<0.05$) MaxPressure to improve mean travel speed by 18%. The control strategy of MaxPressure responds to the maximum product of weighted queue length and saturation flow. The weighted length is the difference between the upstream and downstream queue lengths. Therefore, this control strategy works best if there is an apparent backlog downstream (e.g., waiting vehicles at a neighbouring intersection). In the experimental design, the downstream observation is immediate to outflow links as the communication is local and not coordinated. The 'de facto red' movement could have also contributed to poor performance in travel speed.

DQLA *k-v* vs. DCNN: These controllers have similar DQN structures and action plans. However, they differ in terms of state representation and control strategy. The DTSE for DCNN proved suitable, and earlier analyses showed the superiority of DCNN in signal operation, as in Section 5.1. However, at the network operation level, DCNN performed worse than DQLA *k-v*. Therefore, the outperformance of DQLA *k-v* in terms of time savings and traversing speed is directly associated with the downstream policy. The results

reported that the *k-v* strategy improved waiting time by 16%, travel time by 8%, and mean travel speed by 18% using DQLA *k-v*.

Table 5.3 compares MoPs attributes for the DQLA *k-v* against other systems.

Table 5.3: Measure of performance attributes for comparative logic controllers

MoP	μ Waiting Times (s)		μ Travel Time(s)		μ Travel Speed (m/s)	
Control Logic	**DQLA *k-v* = 125.51**	**Diff. (%)**	**DQLA *k-v* = 398.20**	**Diff. (%)**	**DQLA *k-v* = 3.40**	**Diff. (%)**
Fixed	140.24	-10.50%	420.55	-5.31%	3.40	-0.10%*
Delay	128.51	-2.34%*	452.11	-11.92%	1.21	180.78%
LQFA	198.90	-36.90%	518.63	-23.22%	2.64	28.62%
MaxPressure	122.71	2.28%*	400.31	-0.53%*	2.89	17.73%
DCNN	149.49	-16.04%	434.57	-8.37%	2.88	18.03%

μ: Mean value
*Insignificant difference: p>0.05

5.2.2 Traffic Flow Clearance Ratio

The clearance ratio is the rate of exiting vehicles to entering vehicles in the model environment. Figure 10.1 presents the progression of controllers during the simulation run. The presented data were taken every 13 minutes. The first 13 minutes were disregarded from analyses as the stated duration represents the warm-up period of the model. Overall, all controllers witnessed a steady increase in clearance ratio. The exceptions to this observation are Delay and MaxPressure.

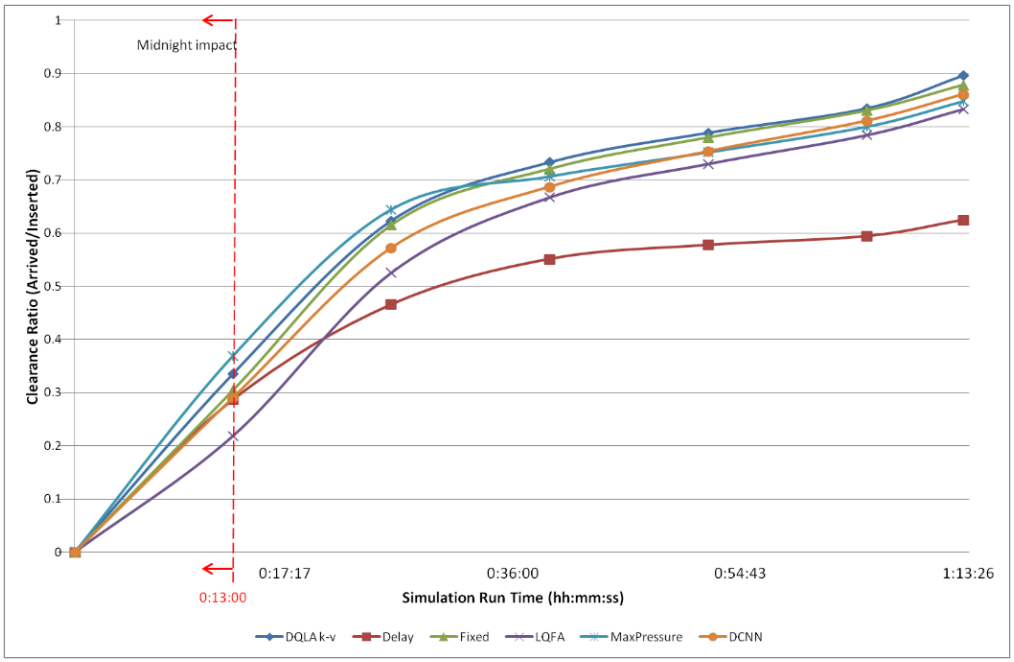


Figure 5.11: Progression of arrived vehicles to inserted vehicles at network level during the test

Delay had the lowest clearance ratio from the beginning of time until the end of the test. The control strategy is imbalanced for closed space and high traffic demand. The actuation technique works best at isolated intersections with travelling directions that have distinct traffic demands. MaxPressure had the highest clearance ratio for the first third of the model run (20 minutes), and then the operation plunged below the proposed DCNN *k-v* and Fixed. At the end of the simulation, MaxPressure finished in 4th place after the DCNN system, indicating that the pressure mechanism could not recover the system. The high arrival rate and defined detection zone caused the deterioration of MaxPressure. As traffic dynamics grew faster and storage capacity was limited, the online MaxPressure policy could not identify the best signal decisions.

The progress of DCNN intersected with MaxPressure in the first half of the peak hour simulation. The result showed the ability of the offline controller to adapt to traffic dynamics. The intelligent controller could make decisions from learned memory on an online test platform. The Fixed system was calibrated offline using a saturation flow rate. Though the Fixed system is rigid, it continued to provide a high clearance ratio. Comparing the policies of DQLA *k*-v and DCNN controllers, the *k*-*v* policy was able to meet flow demands and provide better solutions to signal decisions. The *k*-*v* policy mitigated signal controllers and led to a stable clearance ratio throughout the run time. On the other hand, the halting time of DCNN was less than optimal for mitigating flow when compared to Fixed.

The optimal clearance ratio is 1 ($\frac{exited}{inserted} = 1$). The statistics indicated that DQLA *k*-*v* and Fixed had the highest number of completed vehicle trips compared to other controllers. This saturation flow-based optimisation approach for the Fixed system ensures more phase timings for higher vehicular demands to clear the network and, eventually, a higher arrival rate. On the other hand, the quantitative technique overlooks timing values for road users. The earlier MoPs showed deterioration in waiting time and travel time compared to the DQLA k-v system, as in Section 5.2.1.

The DQLA *k*-*v* achieved a significantly higher arrival rate (~2.30%), although MaxPressure had a higher inserted vehicle (~1.7%) at the network level. The policy for MaxPressure incorporates the product of queue length and density at upstream inbounds, leading to larger storage for routes with

higher traffic demand. In comparison, the proposed DQLA *k-v* mitigates downstream discharges to aim for equilibrium operation (exited/entered=1). The downstream strategy does not adhere to the volume of the inbound traffic stream. As such, the DQLA *k-v* controller restricts the storage capacity for high traffic flow at the upstream approach in favour of utilising discharge zone capacity based on competing traffic turning movements.

The proposed DQLA *k-v* significantly outperformed other signal controllers that aimed to respond to traffic characteristics, including vehicular delay and waiting time, as policy guidance. The responsive systems (LQFA, Delay, and DCNN) can function at a local intersection operation. However, their traffic control techniques cannot achieve global optimisation for single system design and restricted data feeds for network operation. The responsive systems fell back in terms of inserted vehicles (6.70%-9.30%) and arrived vehicles (14.70%-47.90%) to the proposed k-v system policy.

Overall, the DQLA *k-v* had the highest clearance ratio of 0.80 at the end of the peak hour model run. This result indicates that the model operation is close to the optimum operating level under DQLA *k-v*. Table 5.4 presents the mean traffic volumes based on controllers compared to DQLA *k-v*.

Table 5.4: Mean values for traffic input and output at the network level

Traffic Count	**μ Inserted Vehicle Count (veh.)**			**μ Arrived Vehicle Count (veh.)**			**Clearance Ratio**
Control Logic	**DQLA *k*-*v* = 9,827**	**Diff. (%)**	**p-value**	**DQLA *k*-*v* = 7,820**	**Diff. (%)**	**p-value**	**DQLA *k*-*v*** = 0.80
Fixed	10,016	-1.88%	<0.05	7,879	-0.74%	0.64	0.79
LQFA	8,995	9.25%	<0.05	6,597	18.54%	<0.05	0.73
Delay	9,213	6.67%	<0.05	5,287	47.92%	<0.05	0.57
DCNN	8,989	9.33%	<0.05	6,818	14.71%	<0.05	0.76
MaxPressure	9,995	-1.68%	<0.05	7,646	2.28%	<0.05	0.77

5.2.3 Stopped Vehicles

To improve the journey experience, vehicles must traverse to the furthest destination with minimal halts. The relationship between the number of stopped vehicles and logic schemes during the simulation is presented by polynomial functions, as in Figure 5.11.

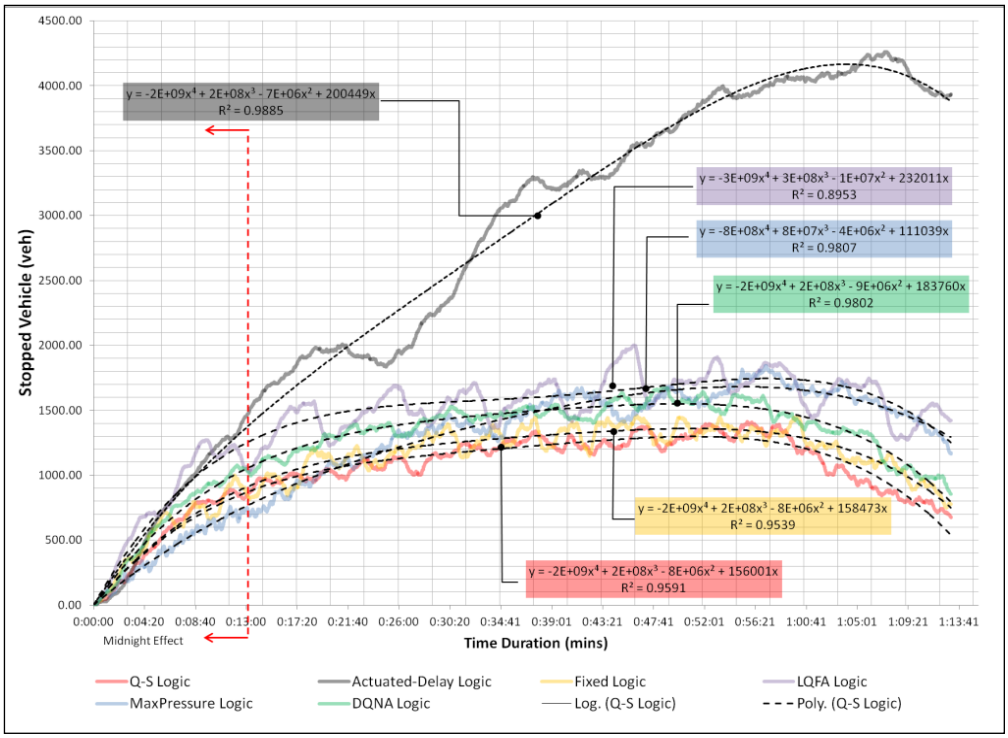


Figure 5.12: Polynomial functions representation for stopped vehicles per logic schemes

At the initial time of network analyses (minute 13), all systems (except Delay) recorded a number of stopped vehicles ≤1,060 vehicles. As the model progressed, the proposed *k-v* strategy maintained the least stopped vehicles at any given time, with a mean value of 1,130 and a standard deviation of 175. The mean value is less than a 7% deviation from the initial simulation record (i.e., 1,060 vehicles). The small standard deviation represents the level of harmony among the independently controlled junctions to handle traffic movement, and the proposed controller is closer to equilibrium than other controllers. The closest function to DQLA *k-v* is the pre-timed controller. The offline optimisation based on saturation flow corresponded well with traffic movement.

In comparison, MaxPressure had started with the least experienced stops, but as inbound flow demand grew, the curve slanted up to the end of traffic volume insertion. At the end of the model run, both online algorithms incorporating queue length elements in the control method (MaxPressure and LQFA) finished with a mean value of 1,498 stopped vehicles. This value represents a 41% increment from the initial simulation experience and a 21% difference from ML and classical system controllers. The actuated system had the worst performance as it could not mitigate the arterial network. The halting vehicle statistical values are presented in Table 5.5.

Table 5.5: Number of halting vehicles statistics

Controller System	Mean (veh.)	Standard Deviation (veh.)	Minimum (veh.)	Maximum (veh.)
Fixed	1,209	164	748	1,460
Delay	3,153	885	1455	4,261
LQFA	1,578	182	1024	2,002
MaxPressure	1,417	267	679	1,844
DCNN	1,379	183	854	1,676
DQLA *k-v*	1,129	175	675	1,414

5.2.4 Network-wide Time Loss

Investigating the travel routes, no single policy ultimately ($p>0.05$) improved the journey experience. This finding is rational, as the stochastic flow nature of traffic means varied arrival rates and the ever-changing dynamics of the traffic environment. The signal function aims to cope with these dynamics and enhance the travelling experience. A comparison between weighted hourly flow and weighted density at the network level was carried out to validate the time loss analyses. The purpose is to ensure that the

experimental settings do not show a significant deviation. The results showed no significant difference ($p>0.05$) between DQLA *k-v* and other controllers in both parameters. The only significant difference was found for DQLA *k-v* and Delay in weighted density. Figure 5.12 presents the weighted flow and density.

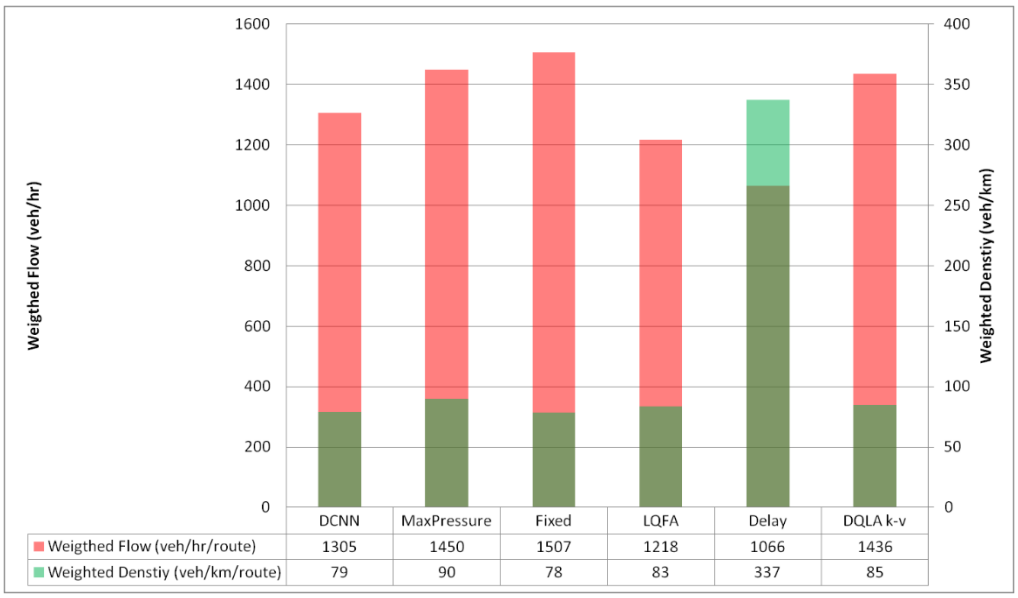


Figure 5.13: Weighted flow and weighted density per route for logic schemes

The mean time loss experienced by a single vehicle on each route is presented in Table 5.6. The recorded data indicated that the proposed DQLA *k-v* has the least average time loss at 98 seconds per route and the least cumulative time loss at 1,560 seconds at the network level. These values present nearly 4% improvements to the nearest rival, i.e., Fixed controller. A travelling vehicle saved about one (1) minute traversing the 7.5km. The time-saving value is crucial for the signal operation and indicates relieved congestion at traversing routes (Bento et al., 2015).

Table 5.6: Time loss based on route and signal controller

Route Links (Junction to Junction)	Mean Time Loss (s)					
	Fixed	**Delay**	**LQFA**	**MaxPressure**	**DCNN**	**DQLA *k-v***
J1-J2	128.63	1290.77	261.46	112.24	94.59	114.46
J2-J3	24.56	1428.65	108.36	23.23	27.81	125.75
J3-J4	3.17	414.78	9.02	4.29	3.68	2.27
J4-J5	12.07	779.73	12.58	100.91	14.82	4.93
J5-J6	35.39	334.89	160.80	567.03	38.18	114.83
J5-J7	165.21	1504.41	86.99	13.84	226.64	8.81
J7-J8	31.08	396.72	42.09	54.49	28.60	16.59
J8-J9	25.92	953.84	17.69	460.22	36.59	13.83
J9-J8	827.31	1178.58	1089.25	895.89	914.01	794.42
J8-J7	1.92	664.09	1.96	1.81	1.88	2.19
J7-J5	10.83	581.26	20.28	135.03	91.31	20.19
J6-J5	77.84	2126.59	65.05	34.12	48.24	115.50
J5-J4	140.99	1081.03	82.98	25.86	74.64	122.78
J4-J3	10.70	1351.41	17.70	3.21	7.87	12.03
J3-J2	25.46	317.57	66.84	50.84	15.80	23.60
J2-J1	97.61	403.70	144.84	63.91	85.04	68.18
Average	**101.17**	**925.50**	**136.74**	**159.18**	**106.86**	**97.52**
Total	**1,618.68**	**14,808.02**	**2,187.90**	**2,546.93**	**1,709.70**	**1,560.35**

Furthermore, the results showed that the offline algorithms (DQLA k-v and DCNN) surpassed the online algorithms (MaxPressure, LQFA, and Delay), producing lower time losses. The measured performance is expected as the applicability of the online algorithm to maintain efficient performance is associated with slow traffic dynamics (Jamshidnejad et al., 2019). In contrast, the offline algorithm is appropriate for high-dimensional features and complex dependencies in the datasets (Cui et al., 2019). Figure 5.13 graphically presents the mean time loss experienced by a single vehicle on each route.

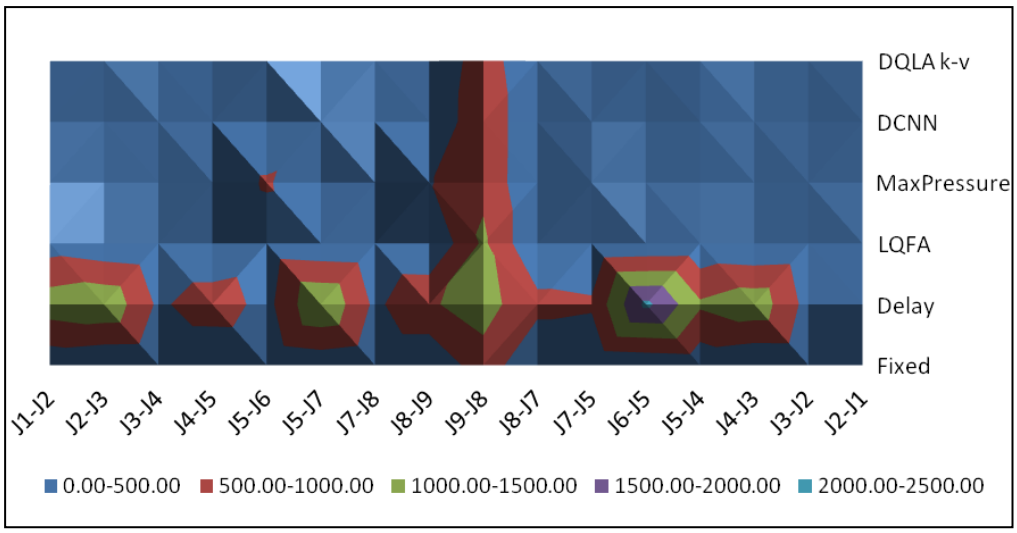


Figure 5.14: Time loss per single vehicle at route links

5.2.5 Closing Remarks for DQLA *k-v* Controller

The network test is carried out using the single-agent system design. Various online and offline logic schemes were used and categorised by generation into pre-time (Fixed), actuated (Delay), and adaptive (LQFA, MaxpPressure, DCNN, and DQLA *k-v*). Each of these systems used a different control strategy. Fixed is an offline optimised controller based on historical saturation flow rate data representing site condition. Delay is an online algorithm incorporating waiting time to manage time plans. LQFA is an online adaptive controller optimising signal operation via queue length at each direction of travel. MaxPressure is an online adaptive controller that manages intersection control based on queue length differences and saturation flow. DCNN utilises offline-taught memory to operate the signal controller using a reward strategy (waiting time). DCNN controller showed superior performance at the isolated intersection, as in earlier Section 5.1. DQLA *k-v*

(similar to the DCNN category) relies purely on a novel downstream available capacity to manage the signal environment. The evaluation of the system was tested in an accurately calibrated arterial network with a high saturation flow (15,508veh/hr) condition.

DQLA *k-v* vs. Comparative Systems: The statistical analyses indicated that the DQLA *k-v* controller improved waiting time between 10% and 36% across different comparative systems, reduced travel time between 5% and 25%, and recorded the highest mean speed at 3.40m/s, a notable 18% to 180% significant improvement.

Furthermore, the DQLA *k-v* logic system had the highest clearance ratio at 80%. This recorded value means eight (8) out of 10 vehicles cleared the network within the peak hour. The optimal flow rate is 1. Indeed, the signalised network cannot reach such an optimum value. Nonetheless, achieving a close-to-optimum score indicates the scalability of the proposed system in mitigating traffic operations.

The vehicles experienced the least number of stops under the guidance of the *k-v* method. In fact, unlike other controllers, the proposed *k-v* technique had a minimal deviation (~7%) from the initial network condition. Moreover, the outbound-based controller significantly reduced experience time loss per route. A vehicle experienced VoT saving between 1 and 10 minutes compared to all other controllers.

Intelligent Control Policies: Adaptive controllers aim to respond to traffic demands and accurately tune the signal operation to meet such demands. The literature review for this generation of controllers indicates that many have focused on system design, but little attention was given to producing a comprehensive policy. The comprehensive policy (i) must not adhere to vehicle characteristics per se, (ii) does not assume vehicle hierarchy, and (iii) mitigates infrastructure corridor. This is because vehicle-based solutions (i) restrict mitigation to specific objective(s) and ignore others, and (ii) have greedy optimisation.

These vehicle-based deficiencies are particularly evident in LQFA and DCNN controllers. Both of these systems relied on control strategies describing the state of vehicles, such as queue length (LQFA) and halting time (DCNN). Generally, the mentioned controllers surpassed the Fixed controller in isolated intersection performance. The findings indicated that LQFA surpassed Fixed in timing attributes as the first controller scheduled longer queues. On the other hand, the scheduling algorithm (LQFA) was not capable of fast convergence at the network level of operation, where mean travel speed and flow rates were typically slower than the Fixed logic.

The DCNN controller managed the isolated intersection better than the Fixed controller. DCNN had a taught memory, which enhanced its experience with environmental dynamics and future mapping characteristics. However, the future prediction of DCNN was not helpful in the arterial network operation. Fixed surpassed DCNN in various traffic attributes. This is not the

first time a DQN controller has failed to operate at the network level. Casas (2017) previously reported that the DQN controller, which optimised operation at an isolated intersection test, had failed to mitigate a network-level operation. The author suggested that more extended training is required to enhance performance. However, the findings in this study suggest a drawback related to control policy. Even if the training claim is correct, a counterclaim is raised for the significant role of policy strategy in enhancing the convergence of DQN to network operation. The DCNN agent was trained for more sessions than the DQLA *k*-*v* agent (Section 4.4.2 Training Agent). The control strategy plays a direct and crucial role in mitigating signal operation.

Delay-based controller failed to operate at the network level. The actuated controllers require noticeable differences between competing directions of travel. At a high arrival rate where all directions tend to have equal or close demands, the controller is crippled to operate. Therefore, control strategies should address the traffic demand hierarchy.

Saturation flow is a component of traffic flow. As a standalone policy, it effectively optimised signal operation at the network level. The pre-timed controller was calibrated based on expert knowledge and was static for a pre-defined period. Signal operations require human interference. In contrast, MaxPressure's approach combined saturation flow and factored queue length. The factor is the difference between the upstream and downstream queue lengths. But the assumption that downstream has a queue length is not always accurate. The backlog of traffic platoon between two (2) intersections is not

always true in the actual environment, as in the tested arterial network. On this basis, MaxPressure becomes the subject of the longest upstream queue, reflecting the highest arrival rate condition. Another way to compute the downstream is by communicating with neighbouring junctions. This study is interested in localised communication to adhere to current infrastructure readiness. Hence, a coordination approach was not attempted and is beyond the scope of this study. Furthermore, coordination is an expensive communication protocol that is challenged in a conflicting work environment.

CHAPTER 6

CONCLUSION

In the past decades, many studies related to adaptive traffic signal controls have been carried out. Various techniques, system designs, and experimental contexts were used for developing the adaptive controls. However, the literature review of over 60 studies showed that the recent research directions are limited to conceptual frameworks and hypothetical assumptions, restricting the proposed methods from practically addressing the control problem. The current proposed adaptive system techniques suffer from **scalability**, **sustenance**, and **valuation** issues.

In order to overcome these major limitations and bring the adaptive control generation a step closer to actual deployment, we designed a single system controller based on the deep reinforcement learning (DRL) technique. The DRL can self-learn to extrapolate the correct action, requires little human intervention, and can solve interactive problems using a reward signal. The design framework of the proposed DRL controller considers the stochastic and dynamic nature of the traffic environment, the low-cost communication protocol using the available detection devices, the scalability factor for network-level operation, and the sustenance of mitigating signal operations.

To illustrate the stochasticity of the traffic environment, the development of traffic micro-models used a real-traffic case study. The illustrative case study consists of nine (9) arterial intersections with highly saturated flow conditions. Two (2) mixed-mode environments were further developed to address the various objectives of this research study. Each vehicle class was modelled, calibrated, and validated to an acceptable accuracy level. Unlike this research, the recent DRL studies do not emphasise traffic flow and environment modelling.

The proposed control agent received data from the junction environment using available built-in devices such as lanearea and induction loops. These detectors were embedded into the traffic micro-model. The detection zones were also restricted to reflect current capability and limit the agent's exposure to dynamics. The detection zones ranged from 70 metres to 140 metres from the stop line, depending on the type of control agent. This design aspect is rarely examined in DRL methods. Instead, the researchers often assume that the signal logic has full exposure to vehicle data. This assumption is based on the utilisation of the futuristic V2X communication channel.

One of the major challenges with adaptive controllers is their capacity to operate at the network level. While studies intended to address the scalability by developing complex multi-agent systems, we resolved the complexity by introducing a state-of-art policy. The policy treats the control problem by mitigating the downstream route's capacity. Unlike vehicle-based

policies (existing DRL studies), the junction-based approach is fairer in regulating flow demands within the intersection level only. No coordination is required for the downstream control policy.

We employed a deep Q-learning algorithm to develop two (2) intelligent controllers: deep convolution neural network (DCNN) and deep Q-learning agent based on density-speed features (DQLA *k*-*v*). Both systems are alike in design architecture but differ in state input and reward function. The design difference comes with the need to evaluate multi-objectives in the study. These developed controllers were tested against different generations of traffic controllers, including fixed, actuated, and adaptive. The training and testing of the proposed DRL controllers were carried out using different traffic data. In contrast to many existing studies, the different training and testing settings are vital for evaluating the system's sustenance.

In the isolated intersection test bed, the DCNN controller performed better than other controllers, particularly in the over-saturated traffic scenario. The system significantly improved travel time between 7% and 38%, reduced waiting time by an average of 90%, and reduced the gap between waiting and moving vehicles from 13% to 43%. In addition, the system showed the highest flow rate at 283veh/hr with the least simulation time. Moreover, the signal logic showed stable performance across various test scenarios. In addition, the DCNN controller showed about 10% higher improvement in waiting time compared to its closest DRL rival system from the literature review.

Regarding the arterial network operation, the DQLA *k-v* proved to surpass other controllers (including the DCNN agent) for network evaluation. The DQLA *k-v* system achieved remarkable performance in optimising network operation. Statistical analyses measured significant cost savings in halting time (10%-36%) and travel time (5%-25%). Moreover, the DQLA *k-v* controller recorded the highest mean travel speed (3.4m/s) compared to other controllers. Consequently, vehicular traffic experienced the least time loss while traversing routes, and it witnessed fewer stops during the trip, leading to close to optimum network operation at a 0.80 clearance ratio.

Overall, it can be concluded that mitigating signal operation using downstream policy is favourable. The development aspect of this control policy can be measured and contained using factors of the physical environment (capacity and available space), including the integration of uncontrolled flow from exclusive turn vehicles, the reliance on built-in infrastructure communication channel, and the limitation of traffic state input to the controller within the specific boundary of a signalised junction.

6.1 Future Work Direction

The findings in this study strongly suggest that the single system design-based DRL method is capable of achieving effective performance in network operation using built-in detectors with a limited amount of data feed within the intersection level. On the other hand, the final product showed the need for an extensive configuration of detectors. The extensive configuration

is expected to induce operational and maintenance costs. This is because loop detectors often breakdown. Further work is required to reduce this reliance and examine the proposed system's actual application.

This work has incorporated a stochastic mixed model evaluation for the intelligent controller and control strategies. Whereas the DRL controllers learnt to optimise the signal operation by forecasting the features of the environment, the impact of vehicular composition was not examined. A future analysis could also examine the impact of various classes of vehicles on operating the adaptive controllers.

Lastly, the analyses and comparisons for the downstream policy enclosed other policies using a local communication protocol. It is recommended that future work broaden the comparison exercise to test the proposed control policy based on local communication against other studies proposing connected vehicle environments for operating the traffic network.

LIST OF REFERENCES

Abdoos, M., Mozayani, N., and Bazzan, A. L., 2013. Holonic multi-agent system for traffic signals control. *Engineering Applications of Artificial Intelligence*, 26 (5-6), pp. 1575-1587, doi: 10.1016/j.engappai.2013.01.007.

Abdulhai, B., Pringle, R., and Karakoulas, G. J., 2003. Reinforcement learning for true adaptive traffic signal control. *Journal of Transportation Engineering*, 129(3), pp. 278-285, doi: https://doi.org/10.1061/(asce)0733-947x(2003)129:3(278).

Ahmed, E. K., Khalifa, A. M., and Kheiri, A., 2018. 'Evolutionary computation for static traffic light cycle optimisation', In *2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*, Khartoum, Sudan, 12-14 August 2018, pp. 1-6. doi: 10.1109/ICCCEEE.2018.8515802.

Al Islam, S. B., and Hajbabaie, A., 2017. Distributed coordinated signal timing optimization in connected transportation networks. *Transportation Research Part C: Emerging Technologies*, *80*, pp. 272-285. doi: 10.1016/j.trc.2017.04.017.

Ali, M. E. M., Durdu, A., Çeltek, S. A., and Yilmaz, A., 2021. An adaptive method for traffic signal control based on fuzzy logic with webster

and modified webster formula using SUMO traffic simulator. *IEEE Access*, *9*, pp. 102985-102997. doi: 10.1109/ACCESS.2021.3094270.

Aslani, M., Megsgari, M. S., and Wiering, M., 2017. Adaptive traffic signal control with actor-critic methods in a real-world traffic network with different traffic disruption events. *Transportation Research Part C: Emerging Technologies* 85, pp. 732-752. doi: 10.1016/j.trc.2017.09.020.

Antoniou, C., Barcelò, J., Brackstone, M., Celikoglu, H., Ciuffo, B., Punzo, V., Sykes, P., Toledo, T., Vortisch, P., and Wagner, P., 2014. Traffic simulation: case for guidelines.

Balaji, P. G, German, X., and Srinivasan, D., 2010. "Urban traffic signal control using reinforcement learning agents", *IET Intelligent Transport Systems*, 4(3), pp. 177-188. doi: 10.1049/iet-its.2009.0096.

Bazzan, A. L., and Klügl, F., 2014. A review on agent-based technology for traffic and transportation. *The Knowledge Engineering Review*, *29*(3), pp. 375-403. doi: 10.1017/S0269888913000118.

Becker, G. S., 1965. A Theory of the Allocation of Time. *The economic journal*, *75*(299), pp. 493-517. doi: 10.2307/2228949.

Bellman, R., 1954. The theory of dynamic programming. *Bulletin of the American Mathematical Society*, *60*(6), pp. 503-515. doi: 10.1090/S0002-9904-1954-09848-8.

Bellemare, M. G., Dabney, W., and Munos, R., 2017. 'A distributional perspective on reinforcement learning'. In *International Conference on Machine Learning*, Sydney, Australia, 6-11 August 2017, pp. 449-458.

Bento, A. M., Roth, K., and Waxman, A., 2015. The value of urgency: Evidence from congestion pricing experiments. Technical report, 2015.

Bi, Y., Srinivasan, D., Lu, X., Sun, Z., and Zeng, W., 2014. Type-2 fuzzy multi-intersection traffic signal control with differential evolution optimization. *Expert systems with applications*, *41*(16), pp. 7338-7349. doi: 10.1016/j.eswa.2014.06.022.

Brys, T., Pham, T. T., and Taylor, M. E., 2014. Distributed learning and multi-objectivity in traffic light control. *Connection Science*, *26*(1), pp. 65-83. doi: 10.1080/09540091.2014.885282.

Budhkar, A. K., and Maurya, A. K., 2017. Characteristics of lateral vehicular interactions in heterogeneous traffic with weak lane discipline. *Journal of Modern Transportation*, *25*(2), pp. 74-89. doi: 10.1007/s40534-017-0130-1.

Casas, N., 2017. *Deep Deterministic Policy Gradient for Urban Traffic Light Control, arXiv.org*. doi: 10.48550/arXiv.1703.09035.

Chin, Y. K., Lee, L. K., Bolong, N., Yang, S. S. and Teo, K. T. K., 2011. 'Exploring Q-learning optimization in traffic signal timing plan management'. In *2011 third international conference on computational intelligence, communication systems and networks*, Bali, Indonesia, 26-28 July 2011, pp. 269-274. doi: 10.1109/CICSyN.2011.64.

Christofa, E., Papamichail, I., and Skabardonis, A., 2013. Person-based traffic responsive signal control optimization. *IEEE Transactions on Intelligent Transportation Systems*, *14*(3), pp. 1278-1289. doi: 10.1109/TITS.2013.2259623.

Chu, K. F., Lam, A. Y., and Li, V. O., 2021. Traffic signal control using end-to-end off-policy deep reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems,* 23(7), pp. 7184-7195. doi: 10.1109/TITS.2021.3067057.

Chu, T., Wang, J., Codecà, L., and Li, Z., 2019. Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE Transactions on Intelligent Transportation Systems*, *21*(3), pp. 1086-1095. doi: 10.1109/TITS.2019.2901791.

Cools, S. B., Gershenson, C., and D'Hooghe, B., 2008. Self-organizing traffic lights: a realistic simulation. In: Advances in Applied Self-organizing Systems. *Advanced Information and Knowledge Processing*. Springer, London. pp. 41–50.

Dai, G., Wang, H., and Wang, W., 2016. Signal optimization and coordination for bus progression based on MAXBAND. *KSCE Journal of Civil Engineering*, *20*(2), pp. 890-898. doi: 10.1007/s12205-015-1516-4.

Cui, Z., Henrickson, K., Ke, R., and Wang, Y., 2019. Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting. *IEEE Transactions on Intelligent Transportation Systems*, *21*(11), pp. 4883-4894. doi: 10.1109/TITS.2019.2950416.

Darmoul, S., Elkosantini, S., Louati, A., and Said, L. B., 2017. Multi-agent immune networks to control interrupted flow at signalized intersections. *Transportation Research Part C: Emerging Technologies*, *82*, pp. 290-313. doi: 10.1016/j.trc.2017.07.003.

De Wit, C. T., and Van Keulen, H., 1987. Modelling production of field crops and its requirements. *Geoderma*, *40*(3-4), pp. 253-265. doi: 10.1016/0016-7061(87)90036-X.

Deligkas, A., Karpas, E., Lavi, R., and Smorodinsky, R., 2018. 'Traffic Light Scheduling, Value of Time, and Incentives'. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)*, Stockholm, Sweden, 13-19 July 2018, pp. 4743-4749.

Dowling, R., Skabardonis, A., and Alexiadis, V., 2004. *Traffic analysis toolbox, volume III: Guidelines for applying traffic microsimulation modeling software* (No. FHWA-HRT-04-040). United States. Federal Highway Administration. Office of Operations.

Dresner, K., and Stone, P., 2004. 'Multiagent traffic management: A reservation-based intersection control mechanism'. In *Autonomous Agents and Multiagent Systems, International Joint Conference on*, July 2004, pp. 530-537.

Ekeila, W., Sayed, T. and Esawey, M.E., 2009. Development of dynamic transit signal priority strategy. *Transportation research record*, *2111*(1), pp. 1-9. doi: 10.3141/2111-01.

El-Ghazali, T., 2020. *Machine learning into metaheuristics: A survey and taxonomy of data-driven metaheuristics*, viewed 05 April 2022, <https://hal.inria.fr/hal-02745295>.

El-Tantawy, S., Abdulhai, B., and Abdelgawad, H., 2013. Multiagent reinforcement learning for integrated network of adaptive traffic signal

controllers (MARLIN-ATSC): methodology and large-scale application on downtown Toronto. *IEEE transactions on Intelligent transportation systems*, 14(3), pp. 1140-1150. doi: 10.1109/TITS.2013.2255286.

Eom, M., and Kim, B., 2020. The traffic signal control problem for intersections: a review, *European Transport Research Review*, 12(1). doi: 10.1186/s12544-020-00440-8.

Feng, Y., Head, K. L., Khoshmagham, S., and Zamanipour, M., 2015. A real-time adaptive signal control in a connected vehicle environment. *Transportation Research* Part C Emerging Technologies. 55, pp. 460–473. doi: 10.1016/j.trc.2015.01.007.

Galvan-Correa, R., Olguin-Carbajal, M., Herrera-Lozada, J. C., Sandoval-Gutierrez, J., Serrano-Talamantes, J. F., Cadena-Martinez, R., and Aquino-Ruíz, C., 2020. Micro artificial immune system for traffic light control. *Applied Sciences*, *10*(21), p. 7933. doi: 10.3390/app10217933.

Gao, K., Zhang, Y., Sadollah, A., and Su, R., 2016. Optimizing urban traffic light scheduling problem using harmony search with ensemble of local search. *Applied Soft Computing*, *48*, pp. 359-372. doi: 10.1016/j.asoc.2016.07.029.

Gao, J., Shen, Y., Liu, J., Ito, M., and Shiratori, N., 2017. Adaptive traffic signal control: Deep reinforcement learning algorithm with experience

replay and target network. *arXiv preprint arXiv:1705.02755*. doi: 10.48550/arXiv.1705.02755.

Gazis, D. C., 1972. Traffic Flow and Control: Theory and Applications: The car increases man's mobility, until all decide to exercise this mobility simultaneously in space and time; then we must call traffic science to the rescue. *American Scientist*, *60*(4), pp. 414-424.

Genders, W., and Razavi, S., 2016. Using a deep reinforcement learning agent for traffic signal control. *arXiv preprint arXiv:1611.01142*. doi: https://doi.org/10.48550/arXiv.1611.01142.

Gershenson, C., 2005. Self-organizing Traffic Lights. *Complex Systems*, 16(1), pp. 29–53. doi: https://doi.org/10.48550/arXiv.nlin/0411066.

Goh, C.Y., Dauwels, J., Mitrovic, N., Asif, M.T., Oran, A., and Jaillet, P., 2012. 'Online map-matching based on hidden markov model for real-time traffic sensing applications', In *2012 15th International IEEE Conference on Intelligent Transportation Systems*, Anchorage, AK, USA, 16-19 September 2012, pp. 776-781. doi: 10.1109/ITSC.2012.6338627.

Gong, Y., Abdel-Aty, M., Cai, Q., and Rahman, M. S., 2019. Decentralized network level adaptive signal control by multi-agent deep reinforcement learning. *Transportation Research Interdisciplinary Perspectives*, *1*, p. 100020. doi: 10.1016/j.trip.2019.100020.

Goodall, N. J., Smith, B. L., and Park, B., 2013. “Traffic Signal Control with Connected Vehicles”, *Transportation Research Record: Journal of the Transportation Research Board*, 2381(1), pp. 65–72. doi: 10.3141/2381-08.

Goodfellow, I., Bengio, Y., and Courville, A., 2017. Deep learning (adaptive computation and machine learning series). *Cambridge Massachusetts*, pp. 321-359.

Gordon, R., and Tighe, W., 2005. *Traffic control systems handbook*. [ebook] Office of Transportation Management Federal Highway Administration, viewed 03 January 2019, <https://ops.fhwa.dot.gov/publications/fhwahop06006/fhwa_hop_06_006.pd>.

Gregoire, J., Qian, X., Frazzoli, E., De La Fortelle, A., and Wongpiromsarn, T., 2014. Capacity-aware backpressure traffic signal control. *IEEE Transactions on Control of Network Systems*, *2*(2), pp. 164-173. doi; 10.1109/TCNS.2014.2378871.

Guler, S. I., Menendez, M., and Meier, L., 2014. "Using connected vehicle technology to improve the efficiency of intersections", *Transportation Research Part C: Emerging Technologies*, 46, pp. 121-131. doi: 10.1016/j.trc.2014.05.008.

Han, M., 2018. Reinforcement learning approaches in dynamic environments. PhD Dissertation, Télécom ParisTech, Paris, France.

Hao, W., Ma, C., Moghimi, B., Fan, Y., and Gao, Z., 2018. Robust optimization of signal control parameters for unsaturated intersection based on tabu search-artificial bee colony algorithm. *IEEE Access*, *6*, pp. 32015-32022. doi: 10.1109/ACCESS.2018.2845673.

Hao, Z., Boel, R., and Li, Z. 2018. "Model based urban traffic control, part I: Local model and local model predictive controllers", *Transportation Research Part C: Emerging Technologies*, 97, pp. 61-81. doi: 10.1016/j.trc.2018.09.026.

Haydari, A, and Yilmaz, Y., 2020, Deep reinforcement learning for intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, pp. 201-216. doi: 10.1109/TITS.2020.3008612.

He, Q., Head, K. L., and Ding, J., 2012. PAMSCOD: Platoon-based arterial multi-modal signal control with online data. *Transportation Research Part C: Emerging Technologies*, *20*(1), pp. 164-184. doi: 10.1016/j.trc.2011.05.007.

Helbing, D., Lämmer, S., and Lebacque, J.P., 2005. Self-organized control of irregular or perturbed network traffic. *Optimal Control and Dynamic Games: Applications in Finance, Management Science and Economics*, pp. 239-274. doi: 10.1007/0-387-25805-1_15.

Hellinga, B. R., 1998. Requirements for the calibration of traffic simulation models. *Proceedings of the Canadian Society for Civil Engineering, 4*, pp. 211-222. doi: 10.3141/1852-17.

Hessel, M., Modayil, J., van Hasselt, H., Schaul, T., Ostrovski, G., Dabney, W., and Silver, D., 2017. Rainbow: Combining improvements in deep reinforcement learning. CoRR, abs/1710.02298. *arXiv preprint arXiv:1710.02298*. doi: 10.1609/aaai.v32i1.11796.

Hnaif, A. A., Nagham, A. M., Abduljawad, M., and Ahmad, A., 2019. 'An intelligent road traffic management system based on a human community genetic algorithm'. In *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT),* Amman, Jordan, 09-11 April 2019, pp. 554-559. doi: 10.1109/JEEIT.2019.8717388.

Horowitz, A., Creasey, T., Pendyala, R., and Chen, M., 2014. *Analytical travel forecasting approaches for project-level planning and design* (No. Project 08-83).

Hunt, P. B., 1982. Split, Cycle and Offset Optimisation Tecnique. *Traffic Engineering & Control*, pp. 190-192.

Hunter-Zaworski, K., Fowler, J., Lall, K., Bardwell, T., Bird, P., and Dahl, S., 2003. Transportation engineering online lab manual. *Oregon State Univ., Corvallis, OR, USA, Tech. Rep*.

Islam, S. M. A. B., and Hajbabaie, A., 2017. Distributed coordination and optimization for signal timing in connected transportation networks, *Transportation Research Part C: Emerging Technologies*, 80, pp. 272–285. doi: 10.1016/j.trc.2017.04.017.

Isukapati, I. K., and Smith, S. F., 2017. 'Accommodating high value-of-time drivers in market-driven traffic signal control'. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, Los Angeles, CA, USA, 1-14 June 2017. pp. 1280-1286. doi: 10.1109/IVS.2017.7995888.

Jamshidnejad, A., Gomes, G., Bayen, A. M., and De Schutter, B., 2019. Integrated Offline and Online Optimization-Based Control in a Base-Parallel Architecture. *arXiv preprint arXiv:1907.05464*. doi: 10.48550/arXiv.1907.05464.

Janssen, S., Andersen, E., Athanasiadis, I. N., and van Ittersum, M. K., 2009. A database for integrated assessment of European agricultural systems. *Environmental Science & Policy*, *12*(5), pp. 573-587. doi: 10.1016/j.envsci.2009.02.005.

Ji, B., Joo, C., and Shroff, N. B., 2012. Delay-based back-pressure scheduling in multihop wireless networks. *IEEE/ACM Transactions on Networking*, *21*(5), pp. 1539-1552. doi: 10.1109/TNET.2012.2227790.

Jiang, T., Wang, Z., and Chen, F., 2021. Urban traffic signals timing at four-phase signalized intersection based on optimized two-stage fuzzy control scheme. *Mathematical Problems in Engineering, 2021*, pp. 1-9. doi: 10.1155/2021/6693562.

Jin, J., and Ma, X., 2017. A group-based traffic signal control with adaptive learning ability, *Engineering applications of artificial intelligence*, 65, pp. 282-293. doi: 10.1016/j.engappai.2017.07.022.

Jin, J., 2018. 'Advance traffic signal control systems with emerging technologies', Doctoral Thesis, KTH Royal Institute of Technology, Stockholm, Sweden.

Jing, P., Huang, H. and Chen, L., 2017. An adaptive traffic signal control in a connected vehicle environment: A systematic review. *Information*, *8*(3), p. 101. doi: 10.3390/info8030101.

Joo, H., Ahmed, S. H., and Lim, Y., 2020. Traffic signal control for smart cities using reinforcement learning. *Computer Communications*, *154*, pp. 324-330. doi: 10.1016/j.comcom.2020.03.005

Kanagaraj, V., Asaithambi, G., Kumar, C. N., Srinivasan, K. K., and Sivanandan, R., 2013. Evaluation of different vehicle following models under mixed traffic conditions. *Procedia-Social and Behavioral Sciences*, *104*, pp. 390-401. doi: 10.1016/j.sbspro.2013.11.132.

Kari, D., Wu, G., and Barth, M. J., 2014. 'Development of an agent-based online adaptive signal control strategy using connected vehicle technology' In *17th international IEEE conference on intelligent transportation systems (ITSC)*, Qingdao, China, 08-11 October 2014, pp. 1802-1807, doi: 10.1109/ITSC.2014.6957954.

Kersebaum, K. C., Boote, K. J., Jorgenson, J. S., Nendel, C., Bindi, M., Frühauf, C., Gaiser, T., Hoogenboom, G., Kollas, C., Olesen, J. E., and Rötter, R. P., 2015. Analysis and classification of data sets for calibration and validation of agro-ecosystem models. *Environmental Modelling & Software, 72*, pp. 402-417. doi: 10.1016/j.envsoft.2015.05.009.

Khamis, M. A., and Gomaa, W., 2014. Adaptive multi-objective reinforcement learning with hybrid exploration for traffic signal control based on cooperative multi-agent framework. *Engineering Applications of Artificial Intelligence, 29*, pp. 134-151. doi: 10.1016/j.engappai.2014.01.007.

Khoo, H. L., 2011. Dynamic penalty function approach for ramp metering with equity constraints. *Journal of King Saud University-Science, 23*(3), pp. 273-279. doi: 10.1016/j.jksus.2010.12.004.

Khoo, H. L., and Tang, C. Y., 2016. Roundabout system capacity estimation and control strategy with origin-destination pattern. *Journal of Transportation Engineering, 142(5)*, p. 04016017. doi: 10.1061/(ASCE)TE.1943-5436.0000838.

Krauß, S., 1998. Microscopic modeling of traffic flow: Investigation of collision free vehicle dynamics.

Krishna K., H., Kumar K. V., and Rao C. H., 2018. Signal design using Webster's method (4 legged intersection). *Indian J. Sci. Res. 17(2)*, pp. 113-119.

Kővári, B., Szőke, L., Bécsi, T., Aradi, S., and Gáspár, P., 2021. Traffic signal control via reinforcement learning for reducing global vehicle emission. *Sustainability*, *13*(20), p. 11254. doi: 10.3390/su132011254.

Lämmer, S., and Helbing, D., 2008. Self-control of traffic lights and vehicle flows in urban road networks. *Journal of Statistical Mechanics: Theory and Experiment*, *2008*(04), p. P04019. doi: 10.1088/1742-5468/2008/04/P04019.

Laval, J., Cassidy, M., and Daganzo, C., 2007. 'Impacts of lane changes at merge bottlenecks: a theory and strategies to maximize capacity'. In *Traffic and Granular Flow'05*, pp. 577-586. Berlin, Heidelberg: Springer.

Lee, J., and Park, B., 2012. Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment. *IEEE transactions on intelligent transportation systems*, *13*(1), pp. 81-90. doi: 10.1109/TITS.2011.2178836.

Li, L., Lv, Y., and Wang, F. Y., 2016. Traffic signal timing via deep reinforcement learning. *IEEE/CAA Journal of Automatica Sinica*, *3*(3), pp. 247-254. doi: 10.1109/JAS.2016.7508798.

Li, L., Huang, W., and Lo, H. K., 2018. Adaptive coordinated traffic control for stochastic demand. *Transportation Research Part C: Emerging Technologies*, *88*, pp. 31-51. doi: 10.1016/j.trc.2018.01.007.

Liang, X., Du, X., Wang, G., and Han, Z., 2019. A deep reinforcement learning network for traffic light cycle control. *IEEE Transactions on Vehicular Technology*, *68*(2), pp.1243-1253. doi: 10.1109/TVT.2018.2890726.

Liu, W., Qin, G., He, Y., and Jiang, F., 2017. Distributed cooperative reinforcement learning-based traffic signal control that integrates V2X networks' dynamic clustering. *IEEE transactions on vehicular technology*, *66*(10), pp. 8667-8681. doi: 10.1109/TVT.2017.2702388.

Liu, R., and Zou, J., 2018. 'The effects of memory replay in reinforcement learning' In *2018 56th annual allerton conference on communication, control, and computing (Allerton)*, Monticello, IL, USA, 02-05 October 2018, pp. 478-485. doi: 10.1109/ALLERTON.2018.8636075.

Lowrie, P. R., 1982. The Sydney co-ordinated adaptive traffic system: Principles, methodology, algorithms.

Ma, W., An, K., and Lo, H. K., 2016. Multi-stage stochastic program to optimize signal timings under coordinated adaptive control. *Transportation Research Part C: Emerging Technologies*, *72*, pp. 342-359. doi: 10.1016/j.trc.2016.10.002.

Ma, J, Fontaine, M. D., Zhou, F., Hu, J., Hale, D. K., and Clements, M. O., 2016. "Estimation of crash modification factors for an adaptive traffic-signal control system", *Journal of Transportation Engineering*, 142(12). doi: 10.1061/(asce)te.1943-5436.0000890

Ma, W., Liu, Y., and Head, K. L., 2014. Optimization of pedestrian phase patterns at signalized intersections: a multi-objective approach. *Journal of advanced transportation*, *48*(8), pp. 1138-1152. doi: 10.1002/atr.1256.

Madrigal Arteaga, V. M., Pérez Cruz, J.R., Hurtado-Beltrán, A., and Trumpold, J., 2022. Efficient Intersection Management Based on an Adaptive Fuzzy-Logic Traffic Signal. *Applied Sciences*, *12*(12), p. 6024. doi: 10.3390/app12126024.

Mannion, P., Duggan, J., and Howley, E., 2015. Parallel reinforcement learning for traffic signal control. *Procedia Computer Science*, *52*, pp. 956-961. doi: 10.1016/j.procs.2015.05.172.

Manual, H.C., 2000. Highway capacity manual. *Washington, DC*, *2*(1).

McKenney, D., and White, T., 2013. “Distributed and adaptive traffic signal control within a realistic traffic simulation”, *Engineering Applications of Artificial Intelligence*, 26(1), pp. 574-583. doi: 10.1016/j.engappai.2012.04.008.

McKay, M. D., Beckman, R. J., and Conover, W. J., 2000. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, *42*(1), pp. 55-61. doi: 10.1080/00401706.2000.10485979.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G. and Petersen, S., 2015. Human-level control through deep reinforcement learning. *nature*, *518*(7540), pp. 529-533. doi: 10.1038/nature14236.

Nafi, N. S., and Khan, J. Y., 2012. ‘A VANET based intelligent road traffic signalling system’, In *Australasian Telecommunication Networks and Applications Conference (ATNAC) 2012*, Brisbane, QLD, Australia, 07-09 November 2012, pp. 1-6. doi: 10.1109/ATNAC.2012.6398066.

Nagabandi, A., Kahn, G., Fearing, R. S., and Levine, S., 2018. ‘Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning’, *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia, 21-25 May 2018, pp. 7559-7566, IEEE. doi: 10.1109/ICRA.2018.8463189.

Oertel, R., and Wagner, P., 2011, January. Delay-time actuated traffic signal control for an isolated intersection. In *Proceedings 90th Annual Meeting Transportation Research Board (TRB)*.

Pandit, K., Ghosal, D., Zhang, H. M., and Chuah, C. N., 2013. Adaptive traffic signal control with vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology*, *62*(4), pp. 1459-1471. doi: 10.1109/TVT.2013.2241460.

Park, B., Won, J., and Perfater, M. A., 2006. *Microscopic simulation model calibration and validation handbook* (No. FHWA/VTRC 07-CR6). Virginia Transportation Research Council.

Pham, T. T., Brys, T., Taylor, M. E., Brys, T., Drugan, M. M., Bosman, P. A., Cock, M. D., Lazar, C., Demarchi, L., and Steenhoff, D., 2013. 'Learning coordinated traffic light control'. In *Proceedings of the Adaptive and Learning Agents workshop (at AAMAS-13)*, St. Paul, USA, 6-10 May 2013, pp. 1196-1201. IEEE.

Płaczek, B., 2014. A self-organizing system for urban traffic control based on predictive interval microscopic model. *Engineering applications of artificial intelligence*, *34*, pp. 75-84. doi: 10.1016/j.engappai.2014.05.004.

Potuzak, T., 2016. 'Optimization of a genetic algorithm for road traffic network division using a distributed/parallel genetic algorithm', In *2016 9th*

International Conference on Human System Interactions (HSI), Portsmouth, UK, 06-08 July 2016, pp. 21-27. doi: 10.1109/HSI.2016.7529603.

Prashanth, L. A., and Bhatnagar, S., 2010. Reinforcement learning with function approximation for traffic signal control. *IEEE Transactions on Intelligent Transportation Systems*, *12*(2), pp. 412-421. doi: 10.1109/TITS.2010.2091408.

Priemer, C.; and Friedrich, B. A., 2009. 'Decentralized adaptive traffic signal control using V2I communication data', *12th International IEEE Conference on Intelligent Transportation Systems*, St. Louis, MO, USA, 04-07 October 2009, pp. 1-6, doi: 10.1109/ITSC.2009.5309870.

Putha, R., Quadrifoglio, L., and Zechman, E., 2012. Comparing ant colony optimization and genetic algorithm approaches for solving traffic signal coordination under oversaturation conditions. *Computer-Aided Civil and Infrastructure Engineering*, *27*(1), pp. 14-28. doi: 10.1111/j.1467-8667.2010.00715.x.

Rasheed, F., Yau, K. L. A., and Low, Y. C., 2020. Deep reinforcement learning for traffic signal control under disturbances: A case study on Sunway city, Malaysia. *Future Generation Computer Systems*, *109*, pp. 431-445. doi: 10.1016/j.future.2020.03.065.

Raphael, J., Maskell, S., and Sklar, E., 2015. ‘From goods to traffic: first steps toward an auction-based traffic signal controller’, *In advances in practical applications of agents, multi-agent systems, and sustainability: the PAAMS collection: 13th International Conference, PAAMS 2015*, Salamanca, Spain, June 3-4, 2015, pp. 187-198. doi: 10.1007/978-3-319-18944-4_16.

Robertson, D. I., 1969. Transyt-a traffic network study tool. rrl report lr 253. *London: TRRL*.

Schepperle, H., and Böhm, K., 2008. ‘July. Auction-based traffic management: Towards effective concurrent utilization of road intersections’, In *2008 10th IEEE Conference on E-Commerce Technology and the Fifth IEEE Conference on Enterprise Computing, E-Commerce and E-Services*, Arlington, VA, USA, 21-24 July 2008, pp. 105-112. doi: 10.1109/CECandEEE.2008.88.

Sacco, N., 2014. Robust optimization of intersection capacity. *Transportation Research Procedia*, *3*, pp. 1011-1020. doi: 10.1016/j.trpro.2014.10.081.

Salter, R. J., and Shahi, J., 1979. Prediction of effects of bus-priority schemes by using computer simulation techniques, *Transportation Research Record*, *718*, pp.1-5.

Shaghaghi, E., Jabbarpour, M. R., Md Noor, R., Yeo, H., and Jung, J. J., 2017. Adaptive green traffic signal controlling using vehicular communication. *Frontiers of Information Technology & Electronic Engineering*, 18, pp. 373-393. doi: 10.1631/FITEE.1500355.

Shen, L., Liu, R., Yao, Z., Wu, W., and Yang, H., 2018. Development of dynamic platoon dispersion models for predictive traffic signal control. *IEEE Transactions on Intelligent Transportation Systems*, *20*(2), pp. 431-440. doi: 10.1109/TITS.2018.2815182.

Shingate, K., Jagdale, K., and Dias, Y., 2020. Adaptive traffic control system using reinforcement learning. *International journal of engineering research and technology, 9(2)*. doi: 10.17577/IJERTV9IS020159.

Smith, S., Barlow, G., Xie, X. F., and Rubinstein, Z., 2013. 'Smart urban signal networks: Initial application of the surtrac adaptive traffic signal control system', In *Proceedings of the International Conference on Automated Planning and Scheduling* , Rome, Italy, 10-14 June 2013, pp. 434-442. doi: 10.1609/icaps.v23i1.13594.

Sokal, R. R., and Rolf F. J., 1995. The principles and practice of statistics in biological research. *Biometry*, pp. 451-554.

Sutton, R. S., and Barto, A. G., 2018. *Reinforcement learning: An introduction*. MIT press.

Tan, T., Bao, F., Deng, Y., Jin, A., Dai, Q., and Wang, J., 2019. Cooperative deep reinforcement learning for large-scale traffic grid signal control. *IEEE transactions on cybernetics, 50*(6), pp. 2687-2700. doi: 10.1109/TCYB.2019.2904742.

Thorpe, T. L., 1997. Vehicle traffic light control using SARSA. Master's Project Rep., Computer Science Dept., Colorado State University, Fort Collins, Colorado, USA.

Thunig, T., Kühnel, N., and Nagel, K., 2019. Adaptive traffic signal control for real-world scenarios in agent-based transport simulations. *Transportation Research Procedia, 37*, pp. 481-488. doi: 10.1016/j.trpro.2018.12.215.

Tiaprasert, K., Zhang, Y., Wang, X. B., and Zeng, X., 2015. Queue length estimation using connected vehicle technology for adaptive signal control. *IEEE Transactions on Intelligent Transportation Systems, 16*(4), pp. 2129-2140. doi: 10.1109/TITS.2015.2401007.

Tunc, I., Yesilyurt, A. Y., and Soylemez, M. T., 2021. Different fuzzy logic control strategies for traffic signal timing control with state inputs. *IFAC-PapersOnLine, 54*(2), pp. 265-270. doi: 10.1016/j.ifacol.2021.06.032.

Urbanik, T., Tanaka, A., Lozner, B., Lindstrom, E., Lee, K., Quayle, S., Beaird, S., Tsoi, S., Ryus, P., Gettman, D., and Sunkari, S., 2015. *Signal timing manual* (Vol. 1). Washington, DC: Transportation Research Board.

Vidali, A., Crociani, L., Vizzari, G., and Bandini, S., 2019. 'A Deep Reinforcement Learning Approach to Adaptive Traffic Lights Management'. In *Workshop "From Objects to Agents"* (*WOA 2019)*, June2019, pp. 42-50.

Varaiya, P., 2013. Max pressure control of a network of signalized intersections. *Transportation Research Part C: Emerging Technologies, 36*, pp. 177-195. doi: 10.1016/j.trc.2013.08.014.

Vasirani, M., and Ossowski, S., 2012. A market-inspired approach for intersection management in urban road traffic networks. *Journal of artificial intelligence research, 43*, pp. 621-659. doi: 10.1613/jair.3560.

Vikhar, P. A., 2016. 'Evolutionary algorithms: A critical review and its future prospects', In *2016 International conference on global trends in signal processing, information computing and communication (ICGTSPICC)*, Jalgaon, India, 22-24 December 2016, pp. 261-265. doi: 10.1109/ICGTSPICC.2016.7955308.

Vilarinho, C., Tavares, J. P., and Rossetti, R. J., 2017. Intelligent traffic lights: Green time period negotiaton. *Transportation research procedia, 22*, pp. 325-334. doi: 10.1016/j.trpro.2017.03.039.

Vogel, K., 2003. A comparison of headway and time to collision as safety indicators. *Accident analysis & prevention*, *35*(3), pp. 427-433. doi: 10.1016/S0001-4575(02)00022-2.

Wan, C. H., and Hwang, M. C., 2018. Value-based deep reinforcement learning for adaptive isolated intersection signal control. *IET Intelligent Transport Systems*, *12*(9), pp. 1005-1010. doi: 10.1049/iet-its.2018.5170.

Wang, P., Jiang, Y., Xiao, L., Zhao, Y., and Li, Y., 2019. A joint control model for connected vehicle platoon and arterial signal coordination. *Journal of Intelligent Transportation Systems*, *24*(1), pp. 81-92. doi: 10.1080/15472450.2019.1579093.

Wang, Y., Ma, W., Yin, W., and Yang, X., 2014. Implementation and testing of cooperative bus priority system in connected vehicle environment: case study in Taicang City, China. *Transportation Research Record*, *2424*(1), pp. 48-57. doi: 10.3141/2424-06.

Wang, S., Xie, X., Huang, K., Zeng, J., and Cai, Z., 2019. Deep reinforcement learning-based traffic signal control using high-resolution event-based data. *Entropy*, *21*(8), p. 744. doi: 10.3390/e21080744.

Wang, Y., Yang, X., Liang, H., and Liu, Y., 2018. “A review of the self-adaptive traffic signal control system based on future traffic

environment", *Journal of Advanced Transportation*, 2018, pp. 1-12. doi: 10.1155/2018/1096123.

Wei, H., Chen, C., Zheng, G., Wu, K., Gayah, V., Xu, K., and Li, Z., 2019. 'Presslight: Learning max pressure control to coordinate traffic signals in arterial network', *In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, Anchorage, USA, 4-8 August 2019, pp. 1290-1298. doi: 10.1145/3292500.3330949.

Wei, H., Zheng, G., Gayah, V., and Li, Z., 2019. A survey on traffic signal control methods. *arXiv preprint arXiv:1904.08117*. doi: 10.48550/arXiv.1904.08117.

Wongpiromsarn, T., Uthaicharoenpong, T., Wang, Y., Frazzoli, E., and Wang, D., 2012. 'Distributed traffic signal control for maximum network throughput', In *2012 15th international IEEE conference on intelligent transportation systems*, Anchorage, AK, USA, 16-19 September 2012, pp. 588-595. doi: 10.1109/ITSC.2012.6338817.

Wunderlich, R. J., 2007. A Longest-Queue-First Signal Scheduling Algorithm with Quality of Service Provisioning for an Isolated Intersection.

Xiang, J., and Chen, Z., 2016. An adaptive traffic signal coordination optimization method based on vehicle-to-infrastructure

communication. *Cluster Computing*, *19*(3), pp. 1503-1514. doi: 10.1007/s10586-016-0620-7.

Xu, J., Yang, K., Shao, Y., and Lu, G., 2015. An experimental study on lateral acceleration of cars in different environments in Sichuan, Southwest China. *Discrete Dynamics in nature and Society, 2015*. doi: 10.1155/2015/494130.

Yau, K. L. A., Qadir, J., Khoo, H. L., Ling, M. H., and Komisarczuk, P., 2017. A survey on reinforcement learning models and algorithms for traffic signal control. *ACM Computing Surveys (CSUR), 50*(3), pp. 1-38. doi: 10.1145/3068287.

Yao, Z., Jiang, Y., Zhao, B., Luo, X., and Peng, B., 2020. A dynamic optimization method for adaptive signal control in a connected vehicle environment. *Journal of Intelligent Transportation Systems*, *24*(2), pp. 184-200. doi:10.1080/15472450.2019.1643723

Yen, C. C., Ghosal, D., Zhang, M. and Chuah, C. N., 2020. 'A deep on-policy learning agent for traffic signal control of multiple intersections', In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, Rhodes, Greece, 20-23 September 2020, pp. 1-6. doi: 10.1109/ITSC45102.2020.9294471.

Yin, B., Dridi, M., El Moudni, A., 2014. Approximate dynamic programming for traffic signal control at isolated intersection. *Advances in*

Intelligent Systems and Computing 285, pp. 369-381, doi: https://doi.org/10.1007/978-3-319-06740-7_31.

Younes, M. B., and Boukerche, A., 2015. Intelligent traffic light controlling algorithms using vehicular networks. *IEEE transactions on vehicular technology*, 65(8), pp. 5887-5899. doi: 10.1109/TVT.2015.2472367.

Zaidi, A. A., Kulcsár, B., and Wymeersch, H., 2016. Back-pressure traffic signal control with fixed and adaptive routing for urban vehicular networks. *IEEE Transactions on Intelligent Transportation Systems*, *17*(8), pp. 2134-2143. doi: 10.1109/TITS.2016.2521424.

Zakariya, A. Y., and Rabia, S. I., 2016. Estimating the minimum delay optimal cycle length based on a time-dependent delay formula. *Alexandria Engineering Journal*, *55*(3), pp. 2509-2514. doi: 10.1016/j.aej.2016.07.029.

Zhang, A., Lipton, Z. C., Li, M., and Smola, A. J., 2021. Dive into deep learning. *arXiv preprint arXiv:2106.11342*. doi: 10.48550/arXiv.2106.11342.

Zhang, M., Ma, J., and Dong, H., 2008. *Developing calibration tools for microscopic traffic simulation final report part II: Calibration framework and calibration of local/global driving behavior and departure/route choice model parameters (No. UCB-ITS-PRR-2008-9)*, viewed 19 August 2023, <https://trid.trb.org/view/873959>.

Zhang, Y., Su, R., Gao, K., and Zhang, Y., 2017. ‘August. Traffic light scheduling for pedestrians and vehicles’, In *2017 IEEE Conference on Control Technology and Applications (CCTA)*, Vancouver, BC, Canada, 24-28 September 2017, pp. 1593-1598. doi: 10.1109/IROS.2017.8206215.

Zhou, L., Wang, Y., and Liu, Y., 2017. Active signal priority control method for bus rapid transit based on Vehicle Infrastructure Integration. *International Journal of Transportation Science and Technology*, *6*(2), pp. 99-109. doi: 10.1016/j.ijtst.2017.06.001.

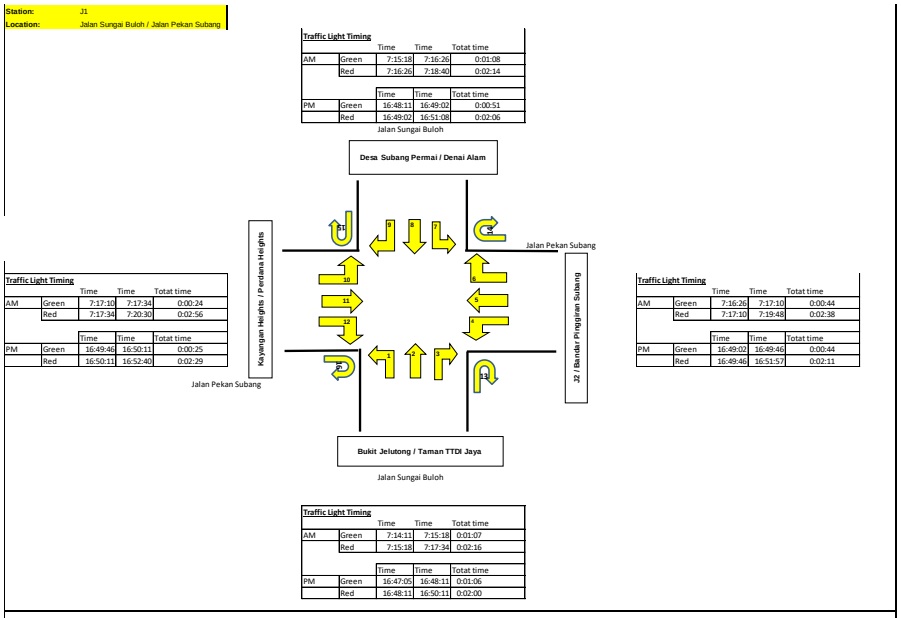
Zeng, J., Hu, J., and Zhang, Y., 2019. ‘Training reinforcement learning agent for traffic signal control under different traffic conditions’, In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, Auckland, New Zealand, 27-30 October 2019, pp. 4248-4254. doi: 10.1109/ITSC.2019.8917342.

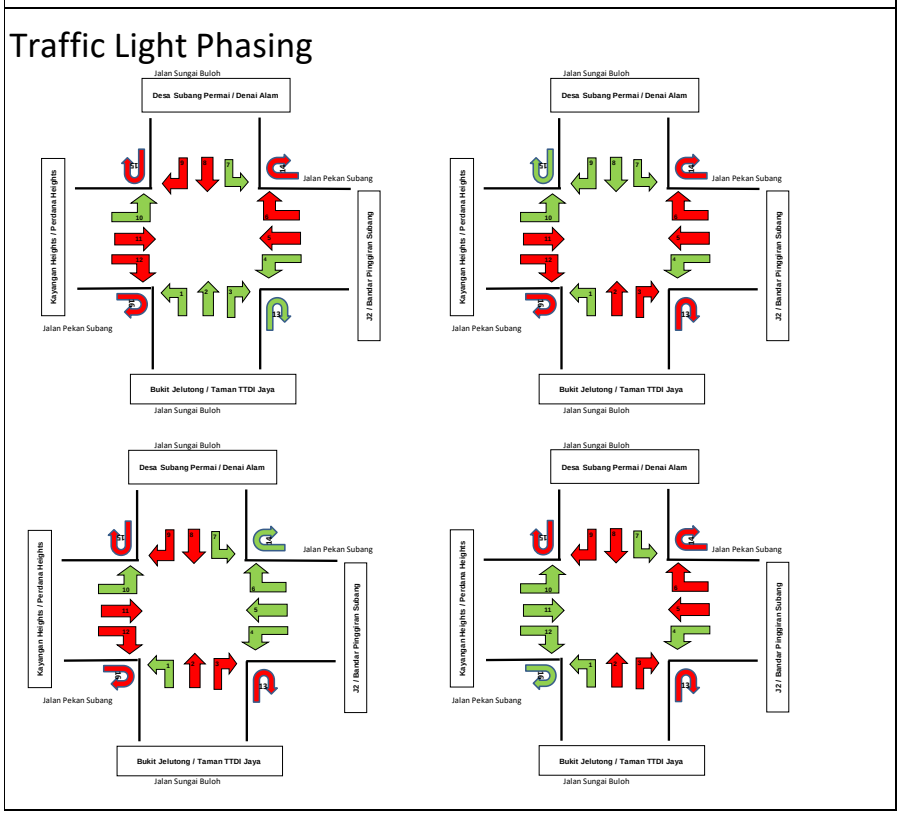
Zeng, X., Zhang, Y., Balke, K. N., and Yin, K., 2014. A real-time transit signal priority control model considering stochastic bus arrival time. *IEEE Transactions on Intelligent Transportation Systems*, *15*(4), pp. 1657-1666. doi: 10.1109/TITS.2014.2304516.

Zheng, J., and Liu, H. X., 2017. Estimating traffic volumes for signalized intersections using connected vehicle data. *Transportation Research Part C: Emerging Technologies*, *79*, pp. 347-362. doi: 10.1016/j.trc.2017.03.007.

APPENDIX A: JUNCTION TIME PLAN AND PHASING

JC1: Junction J1



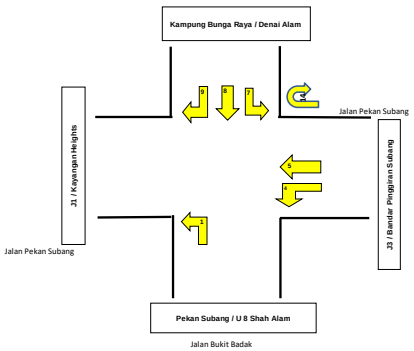


JC2: Junction J2

Station: J2
Location: Jalan Bukit Badak / Jalan Pekan Subang

Traffic Light Timing

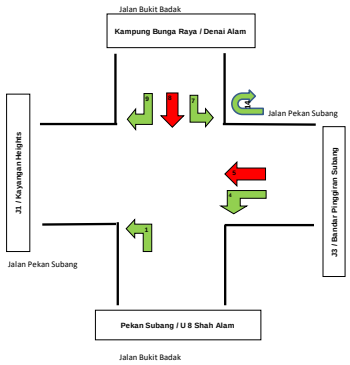
		Time	Time	Totat time
AM	Green	7:22:21	7:22:53	0:00:32
	Red	7:22:53	7:23:56	0:01:03
		Time	Time	Totat time
PM	Green	16:28:53	16:29:31	0:00:38
	Red	16:29:31	16:31:15	0:01:44

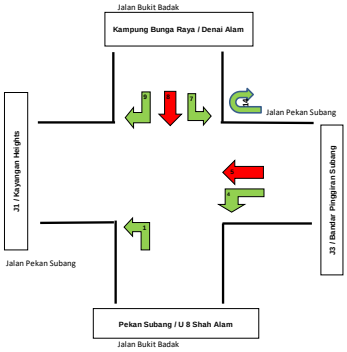


Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:21:20	7:22:21	0:01:01
	Red	7:22:21	7:22:53	0:00:32
		Time	Time	Totat time
PM	Green	16:27:27	16:28:53	0:01:26
	Red	16:28:53	16:29:31	0:00:38

Traffic Light Phasing



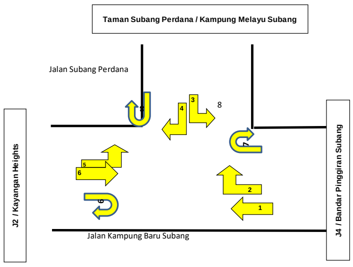


JC3: Junction J3

Station: J3
Location: Jalan Kampung Baru Subang / Jalan Subang Perdana

Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:32:53	7:33:53	0:01:00
	Red	7:33:53	7:35:38	0:01:45
		Time	Time	Totat time
PM	Green	17:02:56	17:04:19	0:01:23
	Red	17:04:19	17:06:48	0:02:29



Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:30:38	7:32:17	0:01:39
	Red	7:32:17	7:33:53	0:01:36
		Time	Time	Totat time
PM	Green	17:00:30	17:02:11	0:01:41
	Red	17:02:11	17:04:19	0:02:08

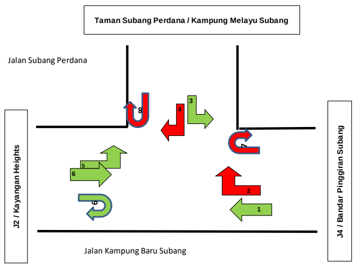
Traffic Light Timing

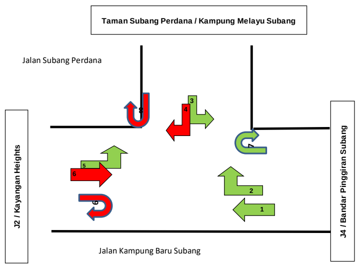
		Time	Time	Totat time
AM	Green	7:30:38	7:32:53	0:02:15
	Red	7:32:53	7:33:53	0:01:00
		Time	Time	Totat time
PM	Green	17:00:30	17:02:56	0:02:26
	Red	17:02:56	17:04:19	0:01:23

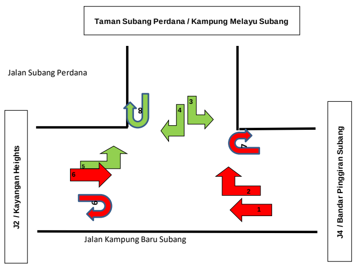
Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:32:17	7:32:53	0:00:36
	Red	7:32:53	7:35:15	0:02:22
		Time	Time	Totat time
PM	Green	17:02:11	17:02:56	0:00:45
	Red	17:02:56	17:05:58	0:03:02

Traffic light Phasing







JC4: Junction J4

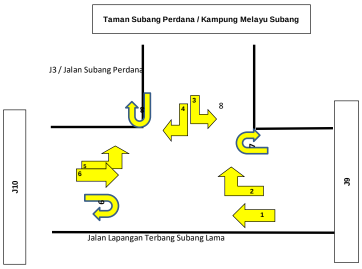
Station: J4
Location: Jalan Lapangan Terbang Subang lama / Jalan Subang Perdana

Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:32:53	7:33:53	0:01:00
	Red	7:33:53	7:35:38	0:01:45
		Time	Time	Totat time
PM	Green	17:02:56	17:04:19	0:01:23
	Red	17:04:19	17:06:48	0:02:29

Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:30:38	7:32:53	0:02:15
	Red	7:32:53	7:33:53	0:01:00
		Time	Time	Totat time
PM	Green	17:00:30	17:02:56	0:02:26
	Red	17:02:56	17:04:19	0:01:23



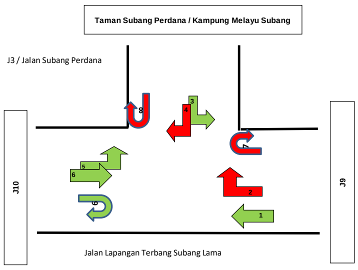
Traffic Light Timing

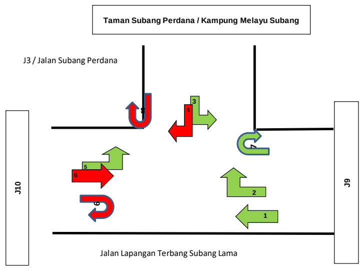
		Time	Time	Totat time
AM	Green	7:30:38	7:32:17	0:01:39
	Red	7:32:17	7:33:53	0:01:36
		Time	Time	Totat time
PM	Green	17:00:30	17:02:11	0:01:41
	Red	17:02:11	17:04:19	0:02:08

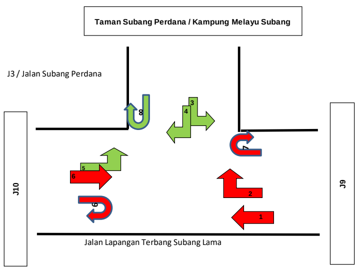
Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:32:17	7:32:53	0:00:36
	Red	7:32:53	7:35:15	0:02:22
		Time	Time	Totat time
PM	Green	17:02:11	17:02:56	0:00:45
	Red	17:02:56	17:05:58	0:03:02

Traffic light Phasing







JC5: Junction J5

Station: J5
Location: Jln Lapangan Terbang Subang Baru / Jln Lapangan Terbang Subang Lama

Direction 1 (Straight Traffic)

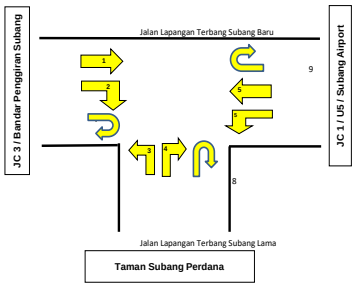
Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:28:42	7:30:40	0:01:58
	Red	7:27:54	7:28:42	0:00:48
		Time	Time	Totat time
PM	Green	17:39:24	17:41:23	0:01:59
	Red	17:38:35	17:39:24	0:00:49

Direction 2 (Right Turn)

Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:30:03	7:30:40	0:00:37
	Red	7:27:54	7:30:03	0:02:09
		Time	Time	Totat time
PM	Green	17:40:45	17:41:23	0:00:38
	Red	17:38:35	17:40:45	0:02:10



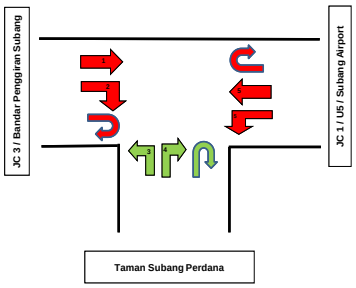
Traffic Light Timing

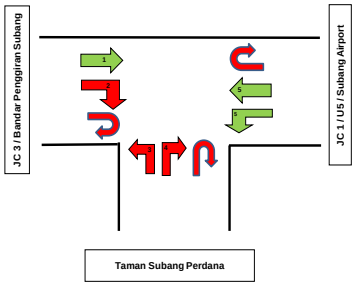
		Time	Time	Totat time
AM	Green	7:28:42	7:30:03	0:01:21
	Red	7:27:54	7:28:42	0:00:48
		Time	Time	Totat time
PM	Green	17:39:24	17:40:45	0:01:21
	Red	17:38:35	17:39:24	0:00:49

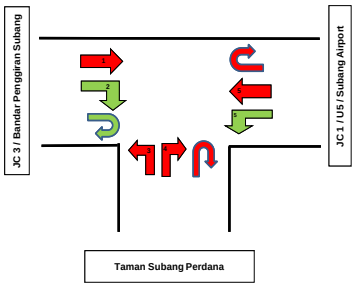
Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:27:54	7:28:42	0:00:48
	Red	7:28:42	7:30:40	0:01:58
		Time	Time	Totat time
PM	Green	17:38:35	17:39:24	0:00:49
	Red	17:39:24	17:41:23	0:01:59

Traffic light Phasing







JC6: Junction J6

Station:	J6
Location:	Jln Lapangan Terbang Subang Baru / Jln Lapangan Terbang Subang Lama

Direction 1 (Straight Traffic)

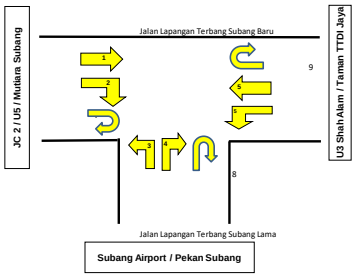
Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:07:32	7:09:27	0:01:55
	Red	7:06:31	7:07:32	0:01:01
		Time	Time	Totat time
PM	Green	17:40:14	17:41:10	0:00:56
	Red	17:39:48	17:40:14	0:00:26

Direction 2 (Right Turn)

Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:09:27	7:09:44	0:00:17
	Red	7:06:03	7:09:27	0:03:24
		Time	Time	Totat time
PM	Green	17:41:10	17:41:23	0:00:13
	Red	17:39:48	17:41:10	0:01:22



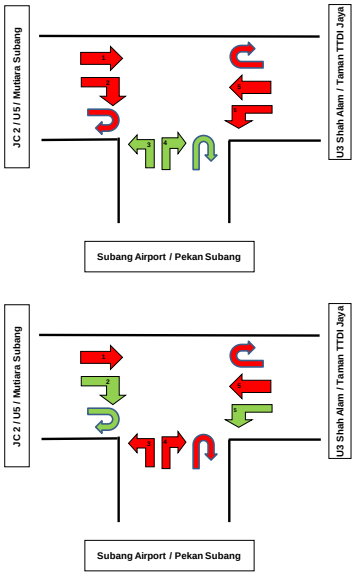
Traffic Light Timing

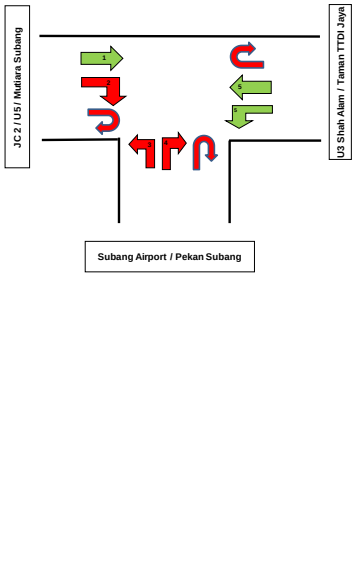
		Time	Time	Totat time
AM	Green	7:07:32	7:09:27	0:01:55
	Red	7:06:31	7:07:32	0:01:01
		Time	Time	Totat time
PM	Green	17:40:14	17:41:10	0:00:56
	Red	17:39:48	17:40:14	0:00:26

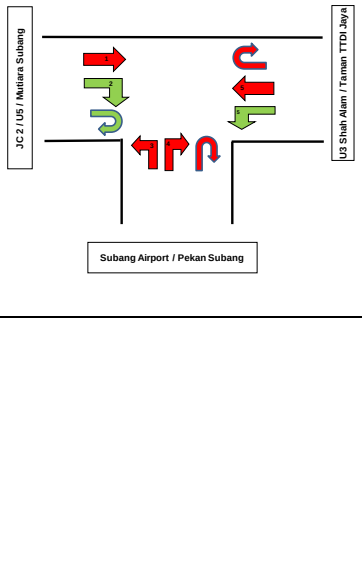
Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:06:31	7:07:32	0:01:01
	Red	7:07:32	7:09:44	0:02:12
		Time	Time	Totat time
PM	Green	17:39:48	17:40:14	0:00:26
	Red	17:40:14	17:41:23	0:01:09

Traffic light Phasing







JC7: Junction J7

Station:	J7
Location:	Jln Lapangan Terbang Subang Baru / Jln Sg Buloh

Direction 1 (Straight Traffic)

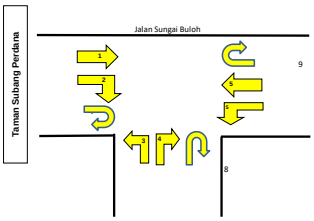
Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:25:11	7:27:31	0:02:20
	Red	7:21:19	7:25:11	0:03:52
		Time	Time	Totat time
PM	Green	17:41:51	17:43:23	0:01:32
	Red	17:38:02	17:41:51	0:03:49

Direction 2 (Right Turn)

Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:25:38	7:27:31	0:01:53
	Red	7:21:19	7:25:38	0:04:19
		Time	Time	Totat time
PM	Green	17:42:20	17:43:23	0:01:03
	Red	17:38:02	17:42:20	0:04:18



Direction 9 (U-Turn)

Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:23:28	7:25:11	0:01:43
	Red	7:21:19	7:23:28	0:02:09
		Time	Time	Totat time
PM	Green	17:40:10	17:41:51	0:01:41
	Red	17:38:02	17:40:10	0:02:08

Direction 6 (Straight Traffic)

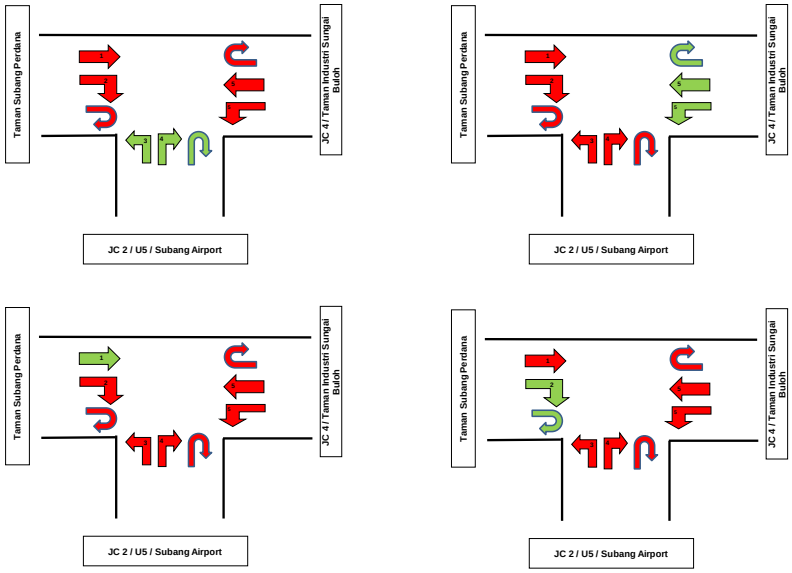
Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:23:28	7:25:38	0:02:10
	Red	7:21:19	7:23:28	0:02:09
		Time	Time	Totat time
PM	Green	17:40:10	17:42:20	0:02:10
	Red	17:38:02	17:40:10	0:02:08

Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:21:19	7:23:28	0:02:09
	Red	7:23:28	7:27:31	0:04:03
		Time	Time	Totat time
PM	Green	17:38:02	17:40:10	0:02:08
	Red	17:40:10	17:43:23	0:03:13

Traffic light Phasing



JC8: Junction J8

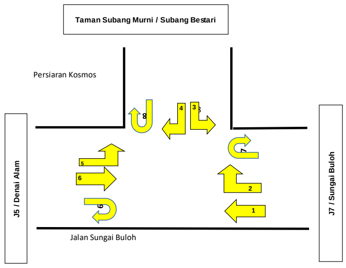
Station: J8
Location: Jalan Sungai Buloh / Persiaran Kosmos

Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:06:59	7:07:43	0:00:44
	Red	7:07:43	7:09:59	0:02:16
		Time	Time	Totat time
PM	Green	16:27:33	16:28:18	0:00:45
	Red	16:28:18	16:30:56	0:02:38

Traffic Light Timing

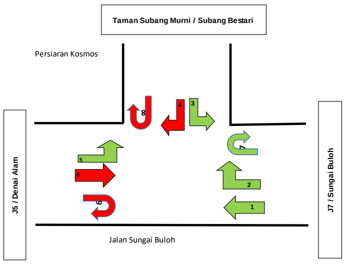
		Time	Time	Totat time
AM	Green	7:07:43	7:09:01	0:01:18
	Red	7:09:01	7:10:45	0:01:44
		Time	Time	Totat time
PM	Green	16:28:18	16:29:40	0:01:22
	Red	16:29:40	16:31:36	0:01:56

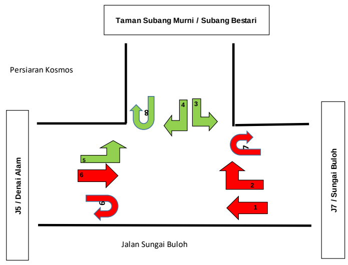


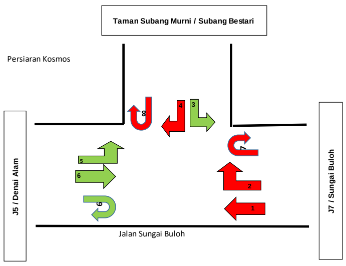
Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:06:03	7:06:59	0:00:56
	Red	7:06:59	7:09:01	0:02:02
		Time	Time	Totat time
PM	Green	16:26:19	16:27:33	0:01:14
	Red	16:27:33	16:29:40	0:02:07

Traffic light Phasing







JC9: Junction J9

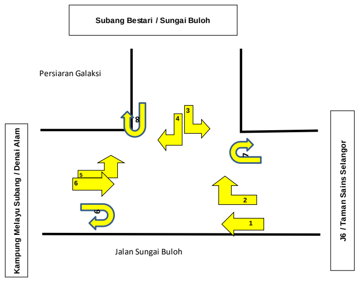
Station:	J9
Location:	Jalan Sungai Buloh / Persiaran Galaksi

Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:17:17	7:17:57	0:00:40
	Red	7:17:57	7:19:11	0:01:14
		Time	Time	Totat time
PM	Green	16:58:42	16:59:15	0:00:33
	Red	16:59:15	17:00:37	0:01:22

Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:16:03	7:17:17	0:01:14
	Red	7:17:17	7:17:57	0:00:40
		Time	Time	Totat time
PM	Green	16:57:22	16:58:42	0:01:20
	Red	16:58:42	16:59:15	0:00:33



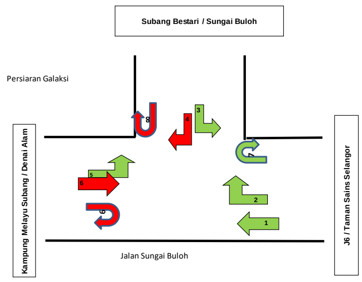
Traffic Light Timing

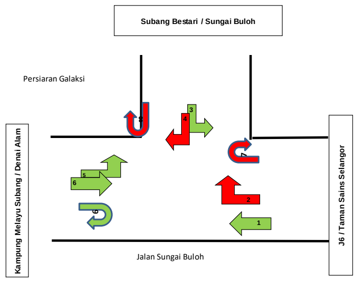
		Time	Time	Totat time
AM	Green	7:16:44	7:17:17	0:00:33
	Red	7:17:17	7:18:37	0:01:20
		Time	Time	Totat time
PM	Green	16:58:03	16:58:42	0:00:39
	Red	16:58:42	16:59:59	0:01:17

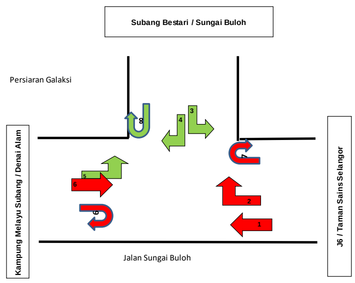
Traffic Light Timing

		Time	Time	Totat time
AM	Green	7:16:03	7:16:44	0:00:41
	Red	7:16:44	7:17:57	0:01:13
		Time	Time	Totat time
PM	Green	16:57:22	16:58:03	0:00:41
	Red	16:58:03	16:59:15	0:01:12

Traffic light Phasing







APPENDIX B: MODEL CALIBRATION AND VALIDATION

Environment Model 1: Isolated Intersection

Calibration is a process to establish input parameter values to reflect traffic conditions (Hellinga, 1998). This is an iterative process until the model parameters are adjusted within a reasonable range.

Some of the parameters have infinite values, as per the SUMO manual documentation. The value definition is presumed to indicate the weight of the parameter in comparison to other factors. Therefore, it is recommended to scale such infinite values to discrete values to make them practice for simulation. To assist with this task, a literature review was carried out to determine the appropriate range for infinite values. The studied modelling parameters include minimum gap to the leading vehicle (minGap), desired headway gap (tau), strategic lane changing (lcStrategic), minimum lateral gap (minGapLat), eagerness for lateral position space (lcSublane), accepting front and rear gaps (lcAssertive).

- minGap (m) is the empty space after a leading vehicle during stops. The default SUMO value is 2.5m. In practice, cars tend to use smaller gaps, especially during congestion. The value for the minGap ranges from 0.8 to 3.2m. The additional value of

0.80m (over the default value) is chosen to allow for motorcycle manoeuvrability.

- tau is a measure of the driver's desired (minimum) time headway. The default SUMO value is 1 second. Vogel (2003) investigated headway. The author found that drivers maintain a constant headway under 2s across various situations. In more complex instances, drivers tend to reduce speed rather than longer headway time. Moreover, smaller headways were measured on the way towards the junction than away from it. This is because the upcoming junction event is a very predictable situation (Vogel, 2003). In Germany, the recommended minimum distance is “half the speedometer”, which means if a car travels at 80 km/h, it should maintain a minimum distance of 40 metres. This rule is translated to a recommended time headway of 1.8 seconds (Vogel, 2003). A similar rule is followed in this study to compute the minimum headway time based on a 90km/hr speed limit. A 45 meters minimum distance is required, equivalent to 1.80 seconds. The upper limit for headway time is 3 seconds (the three-second rule).
- IcStrategic is a value for eagerness to perform strategic lane changing. The default SUMO value is 1. To measure a driver’s eagerness to perform lane-changing across the five (5) vehicle classes, a range of 0 to 4 was tested.

- minGapLat (m) is the desired minimum lateral gap when using the sub-lane model. The default SUMO value is 0.60m. Budhkar and Maurya (2017) studied the characteristics of lateral vehicular interactions in a heterogeneous traffic environment. Based on the authors' findings, the lateral gap tends to be smaller at lower speeds and increases at higher speeds, with motorcycles having the least lateral clearance. The range of lateral gaps is between 0.15 and 2.5 metres.
- IcSublane is the eagerness to use the configured lateral alignment within the lane. The default SUMO value is 1. The higher values result in an increased willingness to sacrifice speed for alignment. A range between 0 (maintain speed) and 4 (sacrifice speed) is used to represent the parameter. The interval values of 0 to 4 weigh the five (5) classes of vehicles.
- lcAssertive (m) represents the willingness to accept lower front and rear gaps on the target lane. The default SUMO value is 1.0. The critical gap is divided by this value. The values between 1 and 5 are used to determine acceptance. The higher the value, the smaller the accepted gap.
- The remaining parameters (sigma, IcImpatience, lcCooperative, IcPushy, and lcLaneDisplane) have values ranging between 0 and 1.

Each of the above 11 parameters was divided at a space interval of 10% (10 clusters). Table 9.1 presents the scalar values for each of the 11 parameters.

Table 9.1: Scalar values for parameterisation exercise

Parameter Value for Simulation											
Parameter ID	**P1**	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**	**P9**	**P10**	**P11**
Actual Frequency (%)	**MinGap (m)**	**sigma**	**tau**	**lcStrategic**	**minGapLat (m)**	**lcSublane**	**lcAssertive (m)**	**lcImpatience**	**lcCooperative**	**lcPushy**	**lcLaneDisplane**
10	0.80	0	1.00	0.00	0.15	0.00	0.15	0.00	0.00	0.00	0.00
20	1.07	0.11	1.22	0.44	0.41	0.44	0.41	0.11	0.11	0.11	0.11
30	1.33	0.22	1.44	0.89	0.67	0.89	0.67	0.22	0.22	0.22	0.22
40	1.60	0.33	1.67	1.33	0.93	1.33	0.93	0.33	0.33	0.33	0.33
50	1.87	0.44	1.89	1.78	1.19	1.78	1.19	0.44	0.44	0.44	0.44
60	2.13	0.56	2.11	2.22	1.46	2.22	1.46	0.56	0.56	0.56	0.56
70	2.40	0.67	2.33	2.67	1.72	2.67	1.72	0.67	0.67	0.67	0.67
80	2.67	0.78	2.56	3.11	1.98	3.11	1.98	0.78	0.78	0.78	0.78
90	2.93	0.89	2.78	3.56	2.24	3.56	2.24	0.89	0.89	0.89	0.89
100	3.20	1.00	3.00	4.00	2.50	4.00	2.50	1.00	1.00	1.00	1.00

Calibration Test Value Sets

The number of possible parameter combinations is equivalent to 9.77×10^6 for the 11 parameters. One method to pull out the most significant scenarios is by using the Latin Hypercube sampling (LHS) method.

The LHS is a statistical method to obtain near-random sample parameter values for a multi-dimensional distribution. The advantage of the

LHS is that it ensures each component is represented in a fully stratified manner, regardless of which components might be significant for the output (McKay et al., 2000). In this regard, the LHS covers the entire parameter surface. This sampling procedure was acquired from Park et al. (2006).

There is one more step to determine the independence of the generated LHS set. The Pearson correlation is used to measure the linear dependency among the variables. The *r* value ranges between 0 and 1. A value nearer to zero (0) indicates no linear relationship, and a value closer to one (1) presents a linear fit. Furthermore, the value of *r* is presented with an arithmetic symbol to indicate a direct relationship (+) and an inverse relationship (-).

Based on Sokal and Rohlf (1995), the significance of the *r* coefficient is dependent on the number of paired variables and the risk value alpha (α). There are 10 parametric sets in this study, or eight (8) degrees of freedom, and a significance value of 0.05. The critical value *r* at which the causation is significant is 0.632. Hence, a significant linear relationship should have a value of 0.632 or more.

The process of producing LHS and verifying the *r* threshold is iterative until an appropriate LHS sample is achieved, as shown in Figure 9.1.

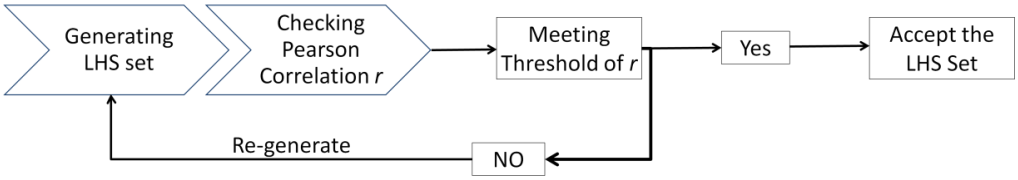


Figure 9.1: Process of generating acceptable LHS set

Table 9.2 presents the 1+20 sets to be tested during calibration. The first set (D1) is the default SUMO parameter set, while the 10 sets correspond to the generated sample space using the LHS method.

Table 9.2: Parameter sets for calibration assignment

Parameter ID											
Value Source	**SUMO Defaults**										
Set ID.	**P1**	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**	**P9**	**P10**	**P11**
D1	2.50	0.50	1.00	1.00	0.60	1.00	1.00	0.00	1.00	0.00	0.00
Value Source	**LHS Sampling Method**										
Set ID.	**minGap (m)**	**sigma**	**tau**	**lcStrategic**	**minGapLat (m)**	**lcSublane**	**lcAssertive (m)**	**lcImpatience**	**lcCooperative**	**lcPushy**	**lcLaneDisplane**
S1	1.33	0.11	2.56	1.78	0.93	0.44	0.67	0.00	0.11	1.00	0.00
S2	2.67	0.22	1.22	3.11	2.24	0.00	0.93	0.67	0.56	0.11	1.33
S3	0.80	0.67	3.00	2.22	1.72	0.89	1.72	0.56	1.00	0.33	2.67
S4	1.60	1.00	1.67	2.67	0.41	1.33	2.50	1.00	0.22	0.78	1.78
S5	1.07	0.44	1.00	1.33	0.67	4.00	0.41	0.44	0.89	0.67	0.44
S6	3.20	0.56	1.89	0.00	0.15	2.22	1.46	0.78	0.78	0.44	2.22
S7	2.13	0.33	2.78	0.44	1.98	2.67	1.19	0.89	0.00	0.22	0.89
S8	2.93	0.78	2.11	3.56	1.19	3.11	0.15	0.33	0.33	0.00	3.11
S9	2.40	0.00	2.33	4.00	1.46	3.56	2.24	0.22	0.67	0.89	3.56
S10	1.87	0.89	1.44	0.89	2.50	1.78	1.98	0.11	0.44	0.56	4.00

The maximum Pearson coefficient is 0.491 (P7-P11) below the threshold of 0.632, which means that sets P11 and P7 are not linearly correlated. The 10 sets and their corresponding parameter values are

acceptable for the calibration test. Table 9.3 presents the Pearson coefficient values.

Table 9.3: Pearson coefficient matrix for parameter ID sets

Parameter ID	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
P1	-0.09	-0.14	0.09	-0.14	0.23	-0.10	0.19	-0.15	-0.39	0.33
P2		-0.23	-0.14	-0.02	-0.04	0.19	0.21	0.05	-0.27	0.34
P3			-0.01	0.11	-0.13	0.06	-0.05	-0.22	-0.12	0.02
P4				0.07	0.04	0.05	-0.21	0.02	0.08	0.36
P5					-0.21	0.10	-0.23	-0.15	-0.39	0.28
P6						-0.16	-0.06	0.12	0.06	0.26
P7							0.21	0.00	0.39	0.49
P8								-0.01	-0.30	-0.22
P9									-0.13	0.25
P10										-0.17

Measure of Performance

A measure of performance (MoP) is applied to identify the performance of certain parameter values. The MoP characterises the distance between the aggregate measurements observed from actual traffic data and the simulation results (Zhang et al., 2008). The link counts are used for the isolated intersection model based on the available data. Three (3) rush hour durations are used from the collected traffic counts to measure the accuracy of the 11 parameter value sets (Table 9.2 above: Parameter sets for calibration assignment).

To measure the simulated traffic counts, detector loops are placed at the end of each approach (before the traffic signal). 10 induction loops were

used to count the through and right-turn movements. The induction loops are placed before the stop line of the modelled junction J1. The detectors record the traffic count based on vehicle class and approach. There are five (5) vehicle classes and four (4) approaches for the modelled junction J1.

The GEH is used to compare the two (2) flow values. The default set D1 was found to be suitable for heavy lorry, and bus modes of transport. The set ID S5 was found to be adequate for the medium lorry class. On the other hand, additional parametric sets are tested for car and motorcycle vehicles.

To improve the GEH for the simulated car vehicle, the value parameter is found by utilising the Pearson coefficient and S5 of parameter values. Additional 10 sets are developed with adjusted values. Table 9.4 presents the set ID S5 parameter values.

Table 9.4: Additional test set driven from S5 parameter values

No.	**min Gap (m)**	**sigma**	**tau**	**lcStrategic**	**minGapLat (m)**	**lcSublane**	**lcAssertive (m)**	**lcImpatience**	**lcCooperative**	**lcPushy**	**lcLaneDisplane**
1	0.80	0	1	0	0.15	0	0.15	0.2	0.5	0.5	0
2	0.89	0.06	1.06	0.22	0.41	0.44	0.41	0.29	0.56	0.56	0.22
3	0.98	0.11	1.11	0.44	0.67	0.89	0.67	0.38	0.61	0.61	0.44
4	1.07	0.17	1.17	0.67	0.93	1.33	0.93	0.47	0.67	0.67	0.67
5	1.16	0.22	1.22	0.89	1.19	1.78	1.19	0.56	0.72	0.72	0.89
6	1.24	0.28	1.28	1.11	1.46	2.22	1.46	0.64	0.78	0.78	1.11
7	1.33	0.33	1.33	1.33	1.72	2.67	1.72	0.73	0.83	0.83	1.33
8	1.42	0.39	1.39	1.56	1.98	3.11	1.98	0.82	0.89	0.89	1.56
9	1.51	0.44	1.44	1.78	2.24	3.56	2.24	0.91	0.94	0.94	1.78
10	1.60	0.50	1.50	2.00	2.50	4.00	2.50	1.00	1.00	1.00	2.00

The following Table 9.5 summarises the 11 calibrated parameters for the SUMO model based on vehicle category.

Table 9.5: Final calibrated parameter values

Parameter ID											
Vehicle Category	**minGap (m)**	**sigma**	**tau**	**lcStrategic**	**minGapLat (m)**	**lcSublane**	**lcAssertive (m)**	**lcImpatience**	**lcCooperative**	**lcPushy**	**lcLaneDisplane**
Passenger Car	0.89	0.22	1.00	0.67	0.67	4.00	0.41	0.44	0.94	0.83	0.22
Motorcycle	0.89	0.22	1.00	0.67	0.67	4.00	0.33	0.44	0.94	0.83	0.22
Medium Lorry	1.07	0.44	1.00	1.33	1.33	4.00	0.41	0.44	0.89	0.67	0.44
Bus/Heavy Lorry	2.50	0.50	1.00	1.00	1.00	1.00	1.00	0.00	1.00	0.00	0.00

The performance measure has shown improvement using the set of parameters in former Table 9.5. The GEH of 5 and below was achieved for two (2) calibration sets, CS2 and CS3, while calibration set CS1 shows a GEH greater than 5. Table 9.6 presents the GEH values for the three (3) calibration data sets.

Table 9.6: GEH value for calibration test sets

Direction		East	North	South	West	Average
Calibration Set	CS1	5.30	13.77	7.78	0.00	7.18
	CS2	1.25	3.88	1.02	3.44	2.78
	CS3	7.86	5.82	0.08	0.00	1.97

The high GEH value is found to be related to the north approach in CS1. Interestingly, the traffic volume in this data set is the highest across all the data sets in this study. The average value of the GEH measure across the three (3) calibration sets is 3.98, which is smaller than the acceptable threshold (GEH<5).

Model Validation

The validation set is utilised in this final step to verify the ability of the model to accurately reproduce data different from those used for calibration.

The error resulting from the validation error is often larger than the calibration error. Nevertheless, a measure performance of 85% is considered good quality, i.e., the link counts should have less than 15% error in them (Dowling et al., 2004).

The following parameter values are used to validate the isolated intersection scenario. The validation set was not used in the earlier calibration procedure. A distinction between the chi-square and GEH methods is that the earlier gives relatively more significant weight to larger differences between flow counts. These differences are reflected in the aggregate measure of computing the average of all GEH statistics from a set of counts. Based on the MoP, the average GEH is equivalent to 2.03 with a 95% confidence level. A total of 64 simulation run are performed to acquire this GEH value during the validation stage. Table 9.7 summarises the GEH based on vehicle type and the junction's approach.

Table 9.7: GEH score for the validation set

GEH						
Approach		**East**	**North**	**South**	**West**	**Average**
Vehicle Category	Car	8.31	5.13	5.84	0.00	**4.82**
	Motorcycle	5.59	1.91	2.76	0.00	**2.56**
	Bus	0.83	0.00	0.00	NA	**0.21**

GEH						
Approach		**East**	**North**	**South**	**West**	**Average**
	Van	2.77	1.13	1.29	0.00	**1.30**
	Lorry	3.96	0.71	0.29	0.00	**1.24**
	Overall Average	**4.29**	**1.78**	**2.04**	**0.00**	**2.03**
NA: No vehicle count (0veh/hr)						

Even though the presented GEH values have shown acceptable MoP for the model, these GEH values exclude the right turn movement at the signalised intersection. Therefore, the overall model accuracy should include traffic counts from all approaches that have been inserted and crossed the screenline position.

Based on the GEH reference, the hourly volume estimates inherently contain more relative error on average in comparison to daily models. On the other hand, the single-hour model should perform better than the daily model, as the earlier model has to encompass a lower variety of conditions. The 1-hour validation set has a total volume of 4,213 vehicles.

The simulation has a runtime of 61 minutes to minimise the impact of the midnight simulation at the beginning of the simulation run. The validation set achieved a 5.95% error. Overall, the hourly simulation has about 94% accuracy, with 4,985 vehicle counts on all approaches successfully inserted and crossed the screenline counting location. This accuracy level is above the minimum required accuracy threshold of 85%, which is set for this study. Therefore, the validation set and its set of parameters are acceptable for testing the model and are suitable for the isolated intersection.

Environment Model 2: Arterial Network Model Validation

Traffic Counts and Flow Volume

The route distribution defines the probability of a vehicle traversing the network routes from its origin to its destination. The network has 13 origins (similarly, 13 destinations) corresponding to its edges. The first step is to ensure that each junction has at least 85% accuracy in terms of link counts. If not, then the route distribution exercise must be revised until the network routes achieve the required road volume and junction volumes to reflect the site condition.

Based on the validation exercise, the network has 93% accuracy in terms of volume at links. Besides that, the GEH mean value is found to be appropriate at 1.46. Table 9.8 presents the 10 runs and their associated volumes per intersection approach.

Table 9.8: Validation results for network model for link counts

	Volume Counts (Veh.)										Average (veh.)	Site Observation (veh.)	Difference (%)	Error Counts <=15% (value of 1) Error Counts >15% (value of 0)	GEH
Run ID	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**					
J1															
North	932	961	938	919	965	969	946	938	951	943	946.2	935	1%	1	0.37
South	1,311	1,315	1,282	1,237	1,286	1,266	1,310	1,287	1,314	1,245	1,285.3	1,299	-1%	1	0.38
East	468	457	480	524	518	522	499	500	540	472	498	476	5%	1	1
West	230	241	227	232	236	221	236	240	228	231	232.2	236	-2%	1	0.25
J2															
North	567	516	568	569	547	540	543	584	525	558	551.7	549	0%	1	0.12
South	506	506	506	506	506	506	506	506	506	506	506	506	0%	1	0
East	1,034	1,000	1,006	1,055	1,019	1,006	1,018	1,035	1,045	1,046	1,026.40	1088	-6%	1	1.89
J3															
North	407	383	388	392	391	381	392	399	420	391	394.4	408	-3%	1	0.68
East	998	1,032	975	1,027	978	1,012	1,000	1,003	1,004	1,021	1,005.00	1025	-2%	1	0.63
West	1,427	1,380	1,347	1,395	1,384	1,381	1,433	1,317	1,403	1,357	1,382.40	1435	-4%	1	1.4
J4															
North	610	595	612	603	601	605	675	590	601	624	611.6	630	-3%	1	0.74
East	789	740	769	775	721	766	794	783	726	799	766.2	702	9%	1	2.37
West	362	312	341	327	341	348	344	350	341	334	340	345	-1%	1	0.27
J5															
North	411	400	417	424	391	435	426	412	414	437	416.7	438	-5%	1	1.03
South	866	868	888	880	840	912	855	837	900	860	870.6	908	-4%	1	1.25
West	589	550	560	583	568	547	594	569	570	545	567.5	589	-4%	1	0.89
J6															
North	1,815	1,851	1,792	1,822	1,838	1,842	1,802	1,803	1,885	1,854	1,830.4	1,878	-3%	1	1.11
South	1,192	1,163	1,185	1,161	1,131	1,194	1,183	1,162	1,159	1,174	1,170.4	1,149	2%	1	0.63
West	331	332	328	327	334	329	330	334	334	332	331.1	331	0%	1	0.01
J7															
South	864	795	883	868	831	903	833	813	858	879	852.7	1,027	-17%	0	5.69
East	1,168	1,140	1,143	1,132	1,133	1,126	1,182	1,155	1,095	1,128	1,140.20	1,189	-4%	1	1.43
West	1,072	1,044	1,059	1,053	1,052	1,073	1,070	1,079	1,059	1,035	1,059.60	1,240	-15%	1	5.32
J8															
North	390	360	367	358	386	402	375	360	395	394	378.7	399	-5%	1	1.03
East	1,225	1,264	1,187	1,161	1,151	1,193	1,209	1,186	1,172	1,110	1,185.80	1,009	18%	0	5.34
West	886	885	863	855	857	905	902	874	879	887	879.3	882	0%	1	0.09
J9															
North	484	516	484	495	486	521	498	522	498	506	501	462	8%	1	1.78
East	1,487	1,484	1,406	1,372	1,365	1,434	1,429	1,418	1,414	1,361	1,417.00	1,268	12%	1	4.07
West	638	683	662	634	638	692	682	683	675	664	665.1	696	-4%	1	1.18
Total											**22,811.50**	**23,099.00**	**-1%**	**93%**	**1.46**

Measure of Performance: Travel Time and Travel Speed

The travel time is the time required for a vehicle to travel between two intersections. This time parameter is computed from the instant the vehicle enters the edge route heading towards its destination junction, and it ends at the instant the vehicle passes its destination junction. Therefore, the travel time might include the waiting time if the vehicle has to wait for its phasing turn movement.

The travel speed parameter is the mean speed of vehicles that travel between two (2) junctions. Like the travel time, the speed is measured from the instant the vehicle enters the edge until it passes its destination junction.

The travel time and speed attributes were recorded from the site based on a few runs using a private passenger car. Based on the validation process, both of these MoPs have reached an acceptable accuracy of 87.50%, which is above the threshold of 85% required for the model. The following Table 9.9 presents the findings for validation.

Table 9.9: Validation results for network model for travel time and speed

Summary of 10 Runs	Mean Travel Time (s)			Mean Speed (m/sec)		
Route (Junction To Junction)	**Model Output**	**Field Data**	**Difference (%)**	**Model Output**	**Field Data**	**Difference (%)**
J1-J2	223.03	199.13	12.00%	9.45	10.74	-12.06%
J2-J3	89.52	97.33	-8.03%	12.96	13.43	-3.46%
J3-J4	80.73	76.27	5.85%	12.42	11.11	11.78%
J4-J5	83.86	88.75	-5.51%	11.99	10.74	11.64%
J5-J6	69.06	76	-9.13%	11.28	12.22	-7.71%
J5-J7	344.8	90.67	280.30%	4.5	10.56	-57.36%
J7-8	125.92	129.17	-2.51%	10.27	11.85	-13.34%
J8-J9	120.09	139.2	-13.73%	9.09	10.42	-12.73%
J9-J8	284.83	182.47	56.10%	6.48	13.33	-51.43%
J8-J7	183.97	189.55	-2.94%	11.81	11.57	2.07%
J7-J5	94.95	97.33	-2.45%	9.93	11.48	-13.47%
J6-J5	64.79	72.68	-10.85%	11.44	10.92	4.79%
J5-J4	225.13	203	10.90%	9.26	9.63	-3.79%
J4-J3	115.42	102.3	12.82%	10.08	10.74	-6.13%
J3-J2	96.36	97.07	-0.73%	11.51	10.37	11.03%
J2-J1	206.57	220.4	-6.27%	9.57	10.33	-7.34%
Total Number of Routes			**16**			**16**
Number of Counts <= 15% Error			**14**			**14**
Number of Counts >= 15% Error			**2**			**2**
Accuracy (%)			**87.50%**			**87.50%**

APPENDIX C: HYPERPARAMETER TUNNING FOR NETWORK MODEL

The following steps were followed to tune the parameters for the network model.

Step1: Determination of Grid Search Scope

Typically, a grid search involves picking values approximately on a logarithmic scale. The learning rate α is taken within the set {0.1, .01, 10^{-3}, 10^{-4}, 10^{-5}} whereas the weight factor γ usually investigated for the following values: 0.5, 0.90, and 0.99 (Goodfellow et al., 2017). During the test runs, the weight value of 0.95 was used instead of 0.99, as it was found to be more suitable for future reward consideration. In total, there are 27 combination sets for the hyperparameters. Table 10.1 presents these various test sets.

Table 10.1: Composition of 27 sets for three parameters of the Q-learning

Discount Factor	Learning Rate	Epsilon	Set ID
0.95	**0.1**	**0.42**	10
		0.44	11
		0.34	12
	0.01	**0.42**	13
		0.44	14
		0.34	15
	0.001	**0.42**	16
		0.44	17
		0.34	18
0.9	**0.1**	**0.42**	1
		0.44	2

Discount Factor	Learning Rate	Epsilon	Set ID
		0.34	3
	0.01	**0.42**	4
		0.44	5
		0.34	6
	0.001	**0.42**	7
		0.44	8
		0.34	9
0.5	**0.1**	**0.42**	19
		0.44	20
		0.34	21
	0.01	**0.42**	22
		0.44	23
		0.34	24
	0.001	**0.42**	25
		0.44	26
		0.34	27

Step 2: Benchmarking Grid Search Results with the Fixed Controller

Step 2 is to shortlist the combinations of appropriate measures based on the fixed controller set. Only the top achievers will be further scrutinised. The performance results are based on 10 frames with a 20 minutes cap time for each combination set ID. Several performance measures were looked at, including the run time, the number of inserted and ended traffic volumes, the waiting time, the mean travel time, and the mean speed. These attributes are generated from SUMO.

Furthermore, two (2) additional attributes were computed, including the ratio of end/inserted vehicles and the ended/loaded vehicles. These two (2) statistics give insight into optimal flow conditions. A balanced traffic flow is associated with a small ratio. Table 10.2 presents the findings.

Table 10.2: Measure of performance for fixed time controller and various DQLA *k-v* set attributes

Controller	Time (hh:mm:ss)	Loaded (veh.)	Inserted (veh.)	Ended (veh.)	ended/inserted (%)	ended/loaded (%)	difference (%)	meanWaitingTime (s)	meanTravelTime (s)	meanSpeed (m/s)	meanSpeedRelative (m/s)
Fixed	0:19:59	5,759	3,900	1,429	37%	25%	12%	38.80	289.91	1.45	0.065
DQLA *k-v* run ID	**Time (hh:mm:ss)**	**Loaded (veh.)**	**Inserted (veh.)**	**Ended (veh.)**	**ended/inserted (%)**	**ended/loaded (%)**	**difference (%)**	**meanWaitingTime (s)**	**meanTravelTime (s)**	**meanSpeed (m/s)**	**meanSpeedRelative (m/s)**
10	0:17:28	5,038	3,678	1,474	39%	29%	11%	34.53	299.98	1.87	0.080
11	0:16:44	4,829	3,820	1,503	39%	31%	8%	27.29	271.82	2.30	0.095
12	0:16:09	4,662	3,447	1,397	40%	30%	10%	39.05	251.69	1.98	0.082
13	0:15:38	4,513	3,518	1,316	37%	29%	8%	26.36	253.26	1.78	0.075
14*	0:17:17	4,981	3,906	1,561	40%	31%	9%	35.54	283.24	1.82	0.076
15	0:14:59	4,328	3,425	1,098	32%	25%	7%	22.80	247.19	1.98	0.086
16*	0:17:33	5,060	4,264	1,743	41%	34%	6%	23.92	265.36	1.92	0.082
17	0:16:44	4,827	3,687	1,234	33%	26%	8%	28.39	283.99	1.68	0.073
18	0:17:18	4,988	4,096	1,608	39%	32%	7%	17.20	244.73	1.84	0.078
1	0:16:49	4,850	3,725	1,351	36%	28%	8%	31.83	283.47	1.89	0.079
2	0:17:44	5,115	4,040	1,577	39%	31%	8%	26.52	283.67	1.83	0.078
3	0:16:56	4,885	3,688	1,410	38%	29%	9%	25.92	263.98	1.78	0.078
4	0:16:28	4,753	3,729	1,418	38%	30%	8%	19.16	256.23	1.70	0.072
5	0:18:13	5,254	4,107	1,782	43%	34%	9%	27.66	286.60	2.11	0.09
6*	0:16:46	4,839	3,915	1,423	36%	29%	7%	22.85	275.43	1.78	0.076
7*	0:19:33	5,634	4,467	1,931	43%	34%	9%	37.90	311.28	2.00	0.082
8	0:17:55	5,167	4,081	1,726	42%	33%	9%	34.13	271.55	2.10	0.09
9*	0:15:19	4,421	3,416	1,198	35%	27%	8%	21.13	260.79	2.22	0.096
19*	0:16:27	4,747	3,662	1,323	36%	28%	8%	31.86	255.88	1.46	0.062
20	0:18:55	5,453	4,132	1,702	41%	31%	10%	31.90	290.26	1.75	0.075
21	0:17:53	5,157	3,648	1,498	41%	29%	12%	32.98	293.46	2.03	0.086
22	0:18:01	5,193	4,090	1,557	38%	30%	8%	36.74	306.51	1.70	0.074
23	0:17:17	4,983	3,808	1,398	36%	28%	9%	28.29	262.81	1.88	0.082
24*	0:16:47	4,842	3,876	1,519	39%	31%	8%	22.74	236.38	1.80	0.079
25	0:16:49	4,853	3,728	1,344	36%	27%	8%	23.48	236.94	1.61	0.07
26*	0:18:55	5,451	4,361	1,816	42%	33%	8%	22.58	265.48	2.01	0.085
27	0:16:17	4,696	3,332	1,281	38%	27%	11%	35.11	251.94	1.90	0.082

*The top performing sets

With careful observation of the results, it was determined that set numbers 14, 16, 6, 7, 9, 19, 24, and 26 achieved the best performances. Therefore, additional testing sets were required to calibrate the decay value (exploration-exploitation) further.

Three (3) epsilon values were found to give exceptional learning results based on the calibration exercise. These values are 0.44, 0.42, and 0.33. The performance indicators are presented in Table 10.3.

Table 10.3: Measure of performance for three epsilon values

epsilon	Loaded (vrh.)	Inserted (veh.)	Ended (veh.)	ended/inserted	ended/loaded	Difference	Mean Waiting Time (s)	Mean Travel Time (s)	Mean Speed (m/s)
0.442	4,687.00	3,542.00	1,370.00	0.39	0.29	0.09	20.43	246.87	2.48
0.420	5,552.00	4,510.00	1,795.00	0.40	0.32	0.07	55.61	265.56	2.02
0.344	5,007.00	3,623.00	1,757.00	0.48	0.35	0.13	24.50	284.52	2.72

Based on the hyperparameter calibration test, the attribute values (Set ID 16) of 0.44, 0.001, and 0.50 for the epsilon, learning rate, and weight, respectively, are found to be most suitable for the Q-learning algorithm.

APPENDIX D: TRAINING AGENT AND MEASURE OF PERFORMANCE

It was impossible to have a smooth, curve-shaped geometry progressing towards convergence. Each episode showed different scoring across performance measures. Hence, a ranking system based on multi-objectives is used to determine a suitable agent for testing. The system is based on the number of halting vehicles, the ratio of clearance, the mean waiting time, the mean travel time, and the mean cruising speed.

Environment Model 1: Isolated Intersection

Training the DCNN Agent: A total of 500 episodes were trained. Table 11.1 presents the measured attributes for every 20 episodes. The top three (3) agents in the following category were then extracted.

1. Highest arrival rate
2. Lowest number of halting vehicles
3. Lowest mean waiting time
4. Lowest mean travel time
5. Highest mean speed, and
6. Highest clearance ratio (arrived vehicles:inserted vehicles)

Table 11.1: Performance measure per training session for DCNN agent for isolated intersection model

Run ID	Arrived (veh.)	Halting (veh.)	Mean Waiting Time (s)	Mean Travel Time (s)	Mean Speed (m/se)	Clearance Ratio
1	2,769.27	95.70	0.276191	89.48972	5.432787	49.1%
20	2,773.336	94.53	0.275374	95.3738	5.44226	49.2%
40	2,773.089	90.99	0.265041	87.72622	5.661022	49.2%
60	2,782.992	84.40	0.266902	85.69429	5.901822	49.4%
80	2,773.614	94.20	0.312645	88.39339	5.64721	49.2%
100	2,748.234	105.94	4.525964	99.75839	5.072495	48.8%
120	2,811.493	97.540	0.315954	92.50248	5.563734	49.9%
140	2,765.583	99.131	0.318511	92.16764	5.444137	49.1%
160	2,815.124	94.824	0.291849	98.64769	5.532718	49.9%
180	2,755.682	109.482	0.26426	103.8559	4.860749	48.9%
200	2,828.104	84.020	0.305298	86.60718	5.92461	50.2%
220	2,759.684	101.419	0.611598	102.0272	5.211448	49.0%
240	2,745.07	110.509	5.345391	112.4887	4.764962	48.7%
260	2,777.374	88.922	0.285544	87.6026	5.804087	49.3%
280	2,777.669	86.663	0.274675	88.90044	5.805981	49.3%
300	2,776.575	88.965	0.273085	96.46739	5.619369	49.3%
320	2,772.398	92.026	0.26894	91.78448	5.483331	49.2%
340	2,767.91	99.594	0.266301	99.86902	5.177281	49.1%
360	2,772.586	94.130	0.272634	93.74133	5.460801	49.2%
380	2,783.555	83.521	0.274415	85.13241	5.94253	49.4%
400	2,772.552	93.161	0.266301	86.507	5.591418	49.2%
420	2,780.567	86.534	0.298948	86.41232	5.869082	49.3%
440	2,774.529	90.528	0.276456	94.85795	5.615191	49.2%
460	2,752.067	98.870	5.104959	112.2856	5.206973	48.8%
480	2,745.014	112.185	0.553615	106.5843	4.825112	48.7%
500	2,769.879	94.778	0.754046	89.97945	5.585552	49.1%

Each of these above measures of performance is weighted equally. A score of one (1) is given to each of the three (3) top scores among the agents. Based on Table 11.2, the highest scorer is agent ID200. The trained agent is most likely to perform best compared to other trained agents for the isolated model.

Table 11.2: Top scoring trained agents for isolated intersection model

Trained Agent	Arrived (veh.)	Halting (veh.)	Mean Waiting Time (s)	Mean Travel Time (s)	Mean Speed (s)	Clearance Ratio	Total Score
200	1	1			1	1	**4**
160	1					1	**2**
120	1					1	**2**
380		1		1	1		**3**
60		1		1	1		**3**
40			1				**1**
180			1				**1**
340			1				**1**
420				1			**1**
Score	**3**	**3**	**3**	**3**	**3**	**3**	**18**

Environment Model 2: Arterial Network

The total number of training episodes for DQLA *k-v* and DCNN agents is 100 and 180 episodes, respectively. Each agent acts solely at the network level and controls one intersection. In other words, the training episode will produce nine (9) agents. The training aims to identify the training session with the highest optimisation impact at the network level. Similar to the isolated intersection, a range of performance measures were taken into account to determine the best training session. Then the associated session's agents will be saved and used for testing. Only the significant results are reported in the following for DQLA *k-v* and DCNN.

Training the DQLA k-v Agent: The training yielded that training session #21 performed significantly better compared to other trained episodes. In particular, this session led to higher arrival rates, shorter travel time, and faster traversing experiences at the network level.

Table 11.3: Measure of performance during the training session of DQLA *k-v* agent for arterial network model

Controller	Session ID #21 = 7299.71					
	Session ID #20	**Session ID #30**	**Session ID #84**	**Session ID #88**	**Session ID #96**	**Session ID #100**
Inserted (veh.)	7206.86*	7076.38	6976.83	7887.91	6697.74	6760.53
Controller	**Session ID #21 = 4674.11**					
	Session ID #20	**Session ID #30**	**Session ID #84**	**Session ID #88**	**Session ID #96**	**Session ID #100**
Arrived (veh.)	4418.19	4279.20	3824.88	4286.89	3311.82	3812.80
Controller	**Session ID #21 = 175.8**					
	Session ID #20	**Session ID #30**	**Session ID #84**	**Session ID #88**	**Session ID #96**	**Session ID #100**
Mean Waiting Time (s)	157.28	194.6	176.02*****	205.76	209.86	178.64*
Controller	**Session ID #21 = 494.93**					
	Session ID #20	**Session ID #30**	**Session ID #84**	**Session ID #88**	**Session ID #96**	**Session ID #100**
Mean Travel Time (s)	530.88	530.71	565.4134	644.48	593.77	559.27
Controller	**Session ID #21 = 2.32**					
	Session ID #20	**Session ID #30**	**Session ID #84**	**Session ID #88**	**Session ID #96**	**Session ID #100**
Mean Speed (m/s)	2.14	2.02	1.77	1.36	1.59	1.72

*****Insignificant at $p>0.05$

Training the DCNN Agent: The DCNN agent was trained earlier for the isolated model. However, as the arterial network has sophisticated dynamics and more complex traffic movement across different junction configurations, training the agent is necessary to capture these variables. Training session ID #104 had the highest score across other training episodes. The following Table 11.4 includes the shortlisted training sessions.

Table 11.4: Measure of performance during the training session of DCNN agent for arterial network model

Session ID #	Inserted (veh.)	Waiting (veh.)	Arrived (veh.)	Halting (veh.)	Mean Waiting Time (s)	Mean Travel Time (s)	Mean Speed (m/s)	Clearance Ratio
104	8,304.78	1,586.29	6,343.57	1,185.18	91.97	387.81	3.74	76.38%
131	8,068.21	1,822.85	5,854.92	1,448.27	90.09	408.40	3.32	72.57%
97	8,047.98	1,843.09	6,104.57	1,235.92	103.07	374.28	3.57	75.85%
69	7,660.72	2,230.34	5,808.89	1,190.96	130.57	375.35	3.64	75.83%

APPENDIX E: STUDENT'S BIOGRAPHY AND LIST OF PUBLICATIONS

Muaid Ahmed received a BEng in Civil Engineering from Swinburne University of Technology in 2013 and a Master degree in Engineering Science from Universiti Tunku Abdul Rahman in 2017. He is currently pursuing a Ph.D (Engineering) degree in traffic and transportation engineering at Universiti Tunku Abdul Rahman. His research interests include traffic management, artificial intelligence, driver behaviour, and accident analyses and prevention. He has several published papers and actively participates in engineering conferences and talks.

The following are the list of publications related to this thesis work.

Ahmed M. A. A., Khoo H. L., and Ng O. E., 2023. Discharge control policy based on density and speed for deep Q-learning adaptive traffic signal, *Transportmetrica B: Transport Dynamics*, 11:1, pp. 1707-1726. doi: 10.1080/21680566.2023.2243388.

Ahmed, M.A.A., Khoo, H. L., and Ng, O. E., 2022. Application of Convolution Neural Network for Adaptive Traffic Controller System. *KSCE Journal of Civil Engineering*, 26(9), pp.4062-4072. doi: 10.1007/s12205-022-1936-x.

Ahmed, M. A. A., Khoo, H. L., and Ng, O. E., 2023. ‘Adaptive Signal Controller based on Convolution Neural Network Agent for Heterogeneous Traffic Environment: A Case Study in Shah Alam, Malaysia’. In proceedings of *The 15th International Conference of Eastern Asia Society for. Transportation Studies (EASTS)*, Selangor, Malaysia, 4-7 September 2023.

Other list of publications related to the student are as follows.

Khoo, K. L., and Ahmed, M. A.., 2018. Modeling of passengers’ safety perception for buses on mountainous roads. *Accident Analysis & Prevention*, 113 (2018), pp. 106-116. doi: 10.1016/j.aap.2018.01.025.

Khoo, K. L., and Ahmed, M. A., 2015. ‘Bus driver behavior on rural highways: a case study of Karak Expressway’. In proceedings of the *Conference of ASEAN Road Safety 2015 (CARS2015)*, Kuala Lumpur: Malaysian Institute of Road Safety Research.

Khoo, K. L., and Ahmed, M. A., 2015. A case study of public bus driver at Batu Feringghi. *Journal of the Eastern Asia Society for Transportation Studies*, 11(2015), pp. 1982-1998. doi: 10.11175/easts.11.1982.