Connected and Automated Vehicle Enabled Traffic Intersection Control with Reinforcement Learning

Gokhan Budan

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Doctor of Philosophy
in
Engineering

Low Carbon Vehicles Research Group
School of Engineering, Computing and Mathematics
Oxford Brookes University

Supervised by:

Prof. Denise Morrey, Prof. Khaled Hayatleh and Dr. Peter Ball

January 2021

blank

blank

Preface

This thesis is submitted for the degree of Doctor of Philosophy at Oxford Brookes University. The research work presented herein was undertaken under the supervision of Prof. Denise Morrey, Prof. Khaled Hayatleh and Dr. Peter Ball. To the best of my knowledge, this research work is original, and references are made for any relevant previous work. This thesis has not been submitted for any other degree at any other university. Part of this research work is published in the following publication which can also be found in Appendix D:

- Budan, G., Hayatleh, K., Morrey, D., Ball, P. and Shadbolt, P., 2018. An analysis of vehicle-to-infrastructure communications for non-signalised intersection control under mixed driving behaviour. *Analog Integrated Circuits and Signal Processing*, *95*(3), pp.415-422

All of the simulation work and practical tests presented in Chapter 5 and Chapter 6 were conducted at Zeta Group in Bicester, UK where I, the author of this thesis, also worked as a full-time employee in the first three quarters of my part-time PhD studies at Oxford Brookes University.

blank

Dedication

I would like to dedicate this thesis to my mother Tülay, my father Mesut, my brother Göktuğ and my wife Louise. Without such loving and caring people around me, this research work would not have been possible for me to complete. They have always supported me in my journey to become one of the leading engineers in intelligent machines and systems as one of my main ambitions in life, and that is principally because I have always felt that it is these engineers and scientists, in academia and industry, that in many ways shape the world. I believe that research has prime importance to understand the complexities involved in developing such systems.

"If one day, my words are against science, choose science."

— Mustafa Kemal Atatürk

blank

Acknowledgements

First and foremost, I am extremely grateful to my esteemed supervisors, Prof. Denise Morrey, Dr. Peter Ball and Prof. Khaled Hayatleh for their invaluable advice, continuous support, and patience during my PhD study. Without their guidance and constant feedback, this PhD work would not have been achievable. I would also like to thank Prof. John Durodola for his guidance and support as the Post Graduate Research Tutor. Additionally, I would like to express my sincere gratitude to Mr. Philip Shadbolt OBE for his treasured support and tutelage throughout my studies, and to Zeta Group, where Mr. Shadbolt is the founder and the CEO, for providing financial funding towards the tuition fees and all the equipment, material and the large warehouse space required for the practical experiments. I truly enjoyed our regular brainstorming sessions in technology in general. My appreciation also goes out to Dr. Umut Genc who is the co-founder and the Managing Director of Eatron Technologies Ltd. in Warwick, UK where I started working as a full-time employee in the final year of my PhD studies. His tremendous understanding and the encouragement of my colleagues has helped immensely for me to establish a healthy work, PhD studies and life balance. Last but not the least, I am thankful to my wife Louise for her unbounded patience and encouragement.

It is also important to mention that COVID-19 coronavirus epidemic broke out in the final year of my PhD studies worldwide, and everyone experienced unprecedented times where life in lockdown affected people all around the world. Wearing face masks, social distancing, support bubbles, track and trace suddenly became reality. Therefore, my heartfelt gratitude and thanks for all those aforementioned people has special importance as on many occasions, their support went above and beyond for me to complete this research work.

blank

Abstract

Recent advancements in vehicle automation have led to a proliferation of studies in traffic control strategies for the next generation of land vehicles. Current traffic signal based intersection control methods have significant limitations on dealing with rapidly evolving mobility, connectivity and social challenges. Figures for Europe over the period 2007-16 show that 20% of road accidents that have fatalities occur at intersections. Connected and Automated Mobility (CAM) presents a new paradigm for the integration of radically different traffic control methods into cities and towns for increased travel time efficiency and safety. Vehicle-to-Everything (V2X) connectivity between Intelligent Transportation System (ITS) users will make a significant contribution to transforming the current signalised traffic control systems into a more cooperative and reactive control system. This research work proposes a disruptive unsignalised traffic control method using a Reinforcement Learning (RL) algorithm to determine vehicle priorities at intersections and to schedule their crossing with the objectives of reducing congestion and increasing safety. Unlike heuristic rule-based methods, RL agents can learn the complex non-linear relationship between the elements that play a key role in traffic flow, from which an optimal control policy can be obtained. This work also focuses on the data requirements that inform Vehicle-to-Infrastructure (V2I) communication needs of such a system. The proposed traffic control method has been validated on a state-of-the-art simulation tool and a comparison of results with a traditional signalised control method indicated an up to 84% and 41% improvement in terms of reducing vehicle delay times and reducing fuel consumption respectively. In addition to computer simulations, practical experiments have also been conducted on a scaled road network with a single intersection and multiple scaled Connected and Automated Vehicles (CAV) to further validate the proposed control system in a representative but cost-effective setup. A strong correlation has been found between the computer simulation and practical experiment results. The outcome of this research work provides important insights into enabling cooperation between vehicles and traffic infrastructure via V2I communications, and integration of RL algorithms into a safety-critical control system.

blank

Table of Contents

List of Figures

List of Tables

List of Symbols

Symbol	Unit	Definition
$\boldsymbol{l_n}$	-	Road lane *n*
$\boldsymbol{t}$	sec	Time instant
$\boldsymbol{d}$	m	Distance
$\boldsymbol{v}$	m/s	Vehicle velocity
$\boldsymbol{\varepsilon}$	-	RL probability of random action in ε-greedy policy
$\boldsymbol{\alpha}$	-	RL learning rate parameter
$\boldsymbol{\beta}$	-	RL step-size parameter
$\boldsymbol{\gamma}$	-	RL discount rate parameter
$\boldsymbol{\lambda}$	-	RL decay rate parameter for eligibility traces
$\boldsymbol{s, s'}$	-	RL current and next states
$\boldsymbol{S_t}$	-	RL state at time *t*
$\boldsymbol{a}$	-	RL action
$\boldsymbol{A_t}$	-	RL action at time *t*
$\boldsymbol{r}$	-	RL reward
$\boldsymbol{R_t}$	-	RL reward at time *t*
$\boldsymbol{\pi}$	-	RL policy, decision-making rule
$\boldsymbol{\pi(s)}$	-	RL action taken in state *s* under deterministic policy π
$\boldsymbol{\pi(a\|s)}$	-	RL probability of taking action *a* in state *s* under stochastic policy π

Nomenclature

Acronym	Definition
5G	Fifth Generation
AES	Advanced Encryption Standard
AI	Artificial Intelligence
AoI	Area of Interest of the traffic intersection
AV	Automated Vehicle
BOM	Bill of Materials
BSM	Basic Safety Message
C-ITS	Cooperative Intelligent Transportation System
CA	Certificate Authority
CAV	Connected Automated Vehicle
CAM	Cooperative Awareness Message
CCAM	Cooperative Connected and Automated Mobility
CCH	Control Channel
CEN	European Committee for Standardization
CHV	Connected Human-Driven Vehicle
CL	Convolutional Layers
CO	Carbon Monoxide
CoA	Control Area of the traffic intersection
COM	Component Object Model
CrA	Critical Area of the intersection
CV	Connected Vehicle
DDPG	Deep Deterministic Policy Gradient
DENM	Decentralized Environmental Notification Message
DLL	Dynamic Link Library
DSRC	Dedicated Short Range Communications
ESC	Electronic Speed Control
ETA	Estimated Time of Arrival
ETSI	European Telecommunications Standards Institute
EVA	Ethylene-Vinyl Acetate
ExA	Exit Area of the traffic intersection
FC	Fully Connected

Acronym	Definition
FCC	Federal Communications Commission
FCFS	First Come First Served traffic control method
fps	Frames Per Second
GLOSA	Green Light Optimised Speed Advisory
GPS	Global Positioning System
GUI	Graphical User Interface
IC	Integrated Circuit
ICA	Intersection Control Agent
IEEE	The Institute of Electrical and Electronics Engineers
IMU	Inertial Measurement Unit
IPv6	Internet Protocol version 6
ITS	Intelligent Transportation System
LSTM	Long Short-Term Memory recurrent neural network
MDP	Markov Decision Process
MILP	Mixed-Integer Linear Programming
MPC	Model Predictive Control
MSE	Mean Square Error
NiCd	Nickel–Cadmium
NiMH	Nickel–Metal Hydride
NOx	Nitrogen Oxide
OBE	On-Board Equipment
OBU	On-Board Unit
OS	Operating System
PHY	Physical Layer
PLA	Polylactic Acid
PPO	Proximal Policy Optimization
PWM	Pulse Width Modulation
QoS	Quality of Service
RC	Remote Controlled
RF	Radio Frequency
RL	Reinforcement Learning
RNN	Recurrent Neural Network
RSU	Roadside Unit
SAC	Soft Actor-Critic

Acronym	Definition
SAE	Society of Automotive Engineers
SCH	Service Channel
SG	Signal Group
SPaT	Signal Phase and Timing
TCT	Trajectory Conflict Table
TD3	Twin Delayed DDPG algorithm
TLC	Traffic Light Control
UML	Unified Modelling Language
VA	Vehicle Agent
V2I	Vehicle to Infrastructure
V2V	Vehicle to Vehicle
V2X	Vehicle to Anything
VOC	Volatile Organic Compounds
VRU	Vulnerable Road User
WAVE	Wireless Access in Vehicular Environments

Chapter 1

1. Introduction

1.1. Motivation

The increasing volume of road transport creates several problems, among which the most serious challenges are congestion growth, rise in energy consumption and carbon emissions (European Commission, 2019b). Energy consumption of the transportation sector including air, road, railway, marine in total accounts for 31% of all sectors in Europe where road transport has the greatest share at 94% of all transport options mainly due to an increase in number of vehicles by 1-2% on average every year since 1990, reaching up to 516 cars per 1000 inhabitants in 2017 (European Commission, 2019a). Therefore, there is an urgent need to address road transport issues to prevent their negative impacts on economy, air quality, journey times and environment.

Technological advancement in recent years, targeting more efficient, safe and sustainable road transport systems, has been instrumental in achieving a paradigm shift when considering the next generation of Intelligent Transportation Systems (ITS). Connected and Automated Vehicles (CAV) and Cooperative ITS (C-ITS) technologies are among these emerging fields that have the potential to radically reshape how our current transport system operates in a response to the aforementioned challenges (Milakis, van Arem and van Wee, 2017). In short, C-ITS involves communication between different users and providers of ITS such as vehicles, traffic infrastructure etc. to establish cooperation by information exchange, while CAV technologies aim to eliminate humans from the driving task in a vehicle.

Traffic management at intersections is one of the most important areas in ITS that can improve the highlighted mobility and congestion issues in a cost-effective way (Mladenovic *et al.*, 2016). The figures in Europe over the period 2007-16 show that 20% of road accidents that have fatalities occur at intersections (European Commission, 2018), and because of safety this puts traffic management at the forefront of areas that need to be addressed as a priority. With penetration of C-ITS and CAV technologies, traffic management methods are expected to harness the power of cooperation as well, in order to better match the traffic demand to the existing infrastructure capacity. Signalised control is the

most common traffic management method worldwide that utilises traffic lights to inform right-of-way. It generally uses static road sensors such as loop detectors, radars and cameras to capture the traffic demand, and as a result, the performance of such systems heavily depends on the number of sensors installed (Atkins, 2016a). Use of C-ITS shared data between the traffic infrastructures and traffic users can overcome this challenge whilst reducing infrastructure costs.

The advancements in computing power and data science has brought a resurgence of Artificial Intelligence (AI) techniques in recent years which are broken down into three main categories: supervised learning, unsupervised learning and Reinforcement Learning (RL) (Sutton and Barto, 2018). Supervised learning deals with finding a relationship between input and output based on example data provided whereas unsupervised learning attempts to find a pattern in a dataset without having any prior knowledge about this data. On the other hand, RL is framed as a sequential decision making paradigm that enables learning by trial-and-error. Among these categories, RL is considered to be the most promising approach for traffic management applications (Haydari and Yilmaz, 2020) mainly due to its ability to deal with stochastic environments such as traffic.

With this in mind, this research project seeks to propose a disruptive traffic control method with no traffic lights, hereinafter referred to as unsignalised traffic control, that integrates C-ITS and CAV features to make better use of the road transport infrastructure. In this method, the intention is to implement RL-based decision making mechanism to manage intersection crossing of vehicles in a proactive way in order to reduce journey times, congestion and improve safety. Unlike a typical Traffic Light Control (TLC) method where each direction of traffic on the road is given right-of-way as a batch of vehicles in turn with traffic lights, in the proposed method, vehicles are assigned with priorities individually by the AI control algorithm, and a dedicated intersection crossing time window is allocated for each vehicle. Another key differentiating point between the two control methods is that TLC systems, in general, are based on complex mathematical traffic models in order to optimise signal timings (Zhao, Dai and Zhang, 2012) for a given intersection whereas the proposed AI based control method in this research work uses a model-free approach in which learning the traffic environment dynamics happens in a trial-and-error way.

1.2. Research Questions

The main research questions are connected to each other and focused on the central research topic in the area of traffic control considering the future of mobility. The questions are formulated as follows:

Q1. How can an AI control strategy improve signalised control based on traffic lights in terms of achieving less congestion, journey time and gas emissions by considering the integration of C-ITS communication features, various levels of CAV penetration rates and different driving behaviours from cautious to assertive?

Q2. What is the impact of the stochastic nature of the traffic environment on how an AI control algorithm is designed? How is the operation of such a control method influenced by the neural network architecture, traffic state representation, reward mechanism and the training strategy?

Q3. How can an AI traffic control method be validated with a generalised framework in both simulation and practical settings? What is the most feasible, representative and cost-effective way to benchmark the performance of the proposed control method against the fixed-time TLC method in order to determine how robust and proactive they are under different traffic scenarios?

1.3. Structure of the Thesis

The structure of the thesis closely follows the research objectives in Section 2.6 and it is divided into 9 chapters including this first chapter where an introduction to the research work is presented. Chapter 2 presents the literature review and necessary background information on wireless communications in ITS, various traffic management approaches at intersections, RL strategies and CAV technologies. Chapter 3 then goes on to outline the wireless communication protocol used between CVs and RSU in conjunction with the data requirements. Chapter 4 presents the details of the RL algorithm for vehicle priority assignment and the methods used for collision avoidance while granting intersection crossing access to vehicles with conflicting trajectories. Chapter 5 explains the concept of algorithm training, in the context of RL, and gives details of the simulation platform and the created traffic environment. Chapter 6 and 7 are concerned with the validation of the proposed traffic control method in the simulation and the scaled testbed scenarios in addition to focusing on the performance comparison of different traffic control methods. Chapter 8 presents the key findings of the research work by analysing the simulation and the testbed experiments results. Chapter 9 draws conclusions on the research work while providing insights and recommendation for future research.

Chapter 2

2. Literature Review

2.1. Vehicle Connectivity and Autonomy

2.1.1. Vehicular Communications in Intelligent Transportation Systems (ITS)

Vehicle connectivity is one of the key enabler technologies for the next generation of ITS infrastructure which will enable information exchange with traffic users to coordinate their actions. This cooperation between different users in traffic is expected to provide significant improvements in road safety and more efficient use of limited road space (European Commission, 2017). Therefore, the European Commission adopted the Cooperative ITS (C-ITS) strategy in November 2016 towards facilitating the regulatory frameworks for the future of land traffic across Europe where Cooperative, Connected and Automated Mobility (CCAM) technologies and services are deployed (European Commision, 2016).

C-ITS Deployment Platform was set up by the European Commission after the C-ITS strategy was adopted as one of the key milestones to develop a shared and synchronised approach for the C-ITS deployment in Europe with all stakeholders (European Commission, 2017), and this platform has been working towards using mature short range communication technology like ITS-G5 (ETSI, 2010a) at 5.9 GHz or wide-area communication technologies like 3G, 4G or 5G. The wireless local area network based on the Institute of Electrical and Electronics Engineers (IEEE) 802.11p/bd standard (IEEE, 2014) is the US market equivalent of ITS-G5 in Europe which offers similar features.

Several applications are supported within the ITS communications architecture in which data can either be transmitted periodically via Cooperative Awareness Messages (CAM) (ETSI, 2014a) or event-based via Decentralized Environmental Notification Message (DENM). DENM is especially important for information dissemination on irregular traffic activities such as road works, hazards and events that has potential impact on road safety (ETSI, 2010b).

CAM messages are transmitted in a single hop communication. Therefore, direct communication range is required to exchange CAM data. Unlike CAM, DENM can be disseminated to other users within the same local geographical area in a multi-hop communication method (ETSI, 2010b). The data exchange frequency for mobile users such as vehicles can be between 10 Hz and 1 Hz whereas it can be greater than 1Hz for static stations such as Road Side Units (RSU) (ETSI, 2014b).

A study by Asselin-Miller *et al.*, (2016) on the deployment of C-ITS in Europe reports that using cellular networks like 5G can accelerate the deployment of these systems greatly as the underlying technology has already been implemented and in use today. However, uncertainties exist around the latency times that can be experienced for safety-critical applications when this technology is used. The authors also highlight that the benefits of C-ITS can be significant especially in urban areas however, there is not enough data to measure the effectiveness of some applications including traffic control. With this in mind, it is expected that this research work will provide data in this domain.

2.1.2. Vehicle-to-Infrastructure (V2I) Communications

Within ITS, the Vehicle-to-Everything (V2X) umbrella term is used to represent wireless communications between vehicles and all other users. Infrastructure is one of these users and Vehicle-to-Infrastructure (V2I) communications has great importance for traffic control applications which will be reviewed in this section.

Vehicle On-Board Unit (OBU) and RSU are two main components in a V2I architecture for traffic control. Varga et al. (2017) implements a proof of concept system in the field that conforms to the C-ITS standards (See Section 2.1.1). Vehicles receive broadcasted local MAP (physical intersection geometry data) and Signal Phase and Time (SPaT) data from the RSU and transmit CAM data for intersection crossing. The feasibility of the proposed solution is demonstrated for various environments including rural, highway and urban traffic. On the other hand, Parra et al., 2017 argue that communication reliability becomes a challenge in busy urban traffic scenarios and the authors suggest this could be solved by improving the channel usage mechanism for heavily congested traffic scenarios.

Steinmetz et al. (2014) study V2I communication performance for an intersection control problem. The main objective of the study is to determine the optimum uplink and downlink parameters to ensure reliable communication. The simulation results show that a minimum of 100 uplink channels and relatively high uplink power are required for a reasonable quality of service based on 100 ms of vehicle

data broadcast interval. In terms of downlink, the authors found that a time interval of 300-400ms is the most appropriate value to disseminate vehicle coordination data from the intersection controller. In the same vein, Lee and Park (2015) found out that when there are less than 30 vehicles in the communication zone of 150m, the packet drop is 0.01% which is insignificant.

There is also a growing body of literature that recognises the problems associated with redundant, irrelevant, large data exchange between ITS users. Drawing on an extensive range of sources, Rettore et al., 2019 focuses on the concept of vehicular data space in which V2X communication scenarios are described from data point of view. It is argued that for a traffic control scenario, acquiring data on vehicle states, weather and road conditions are fundamental.

2.1.3. Connected and Automated Vehicles (CAV)

The driving task in land transportation requires a complex skillset for safe and efficient movement of vehicles between locations. Autonomous vehicles must be equipped with sufficient tools to be able to handle unpredictable situations and make timely and safe decisions. Even though the state-of-the-art CAV solutions today cannot yet achieve the performance level that is acceptable for unsupervised driving under any condition and environment (Schwarting, Alonso-Mora and Rus, 2018), the automotive industry has witnessed a rapid progress recently on CAV technologies with cost reduction of vehicle sensors and computing resources together with an increase in their availability (Pendleton et al., 2017).

The Society of Automotive Engineers (SAE) provides a taxonomy that describes a range of vehicle automation levels between SAE Level 0-5, and the autonomous driving capabilities increase as the level number increases (SAE International, 2018). For example, SAE Level 5 vehicle refers to a fully autonomous vehicle where there is no human driver, and the vehicle has the capability to drive everywhere on the road network under all conditions.

The CAV technology is expected to disrupt many aspects of our mobility and lives in general. The sequential effects of CAVs on mobility and society from short to long term are shown in three steps in Figure 1 based on the rippled effect diagram by Milakis, van Arem and van Wee (2017). Congestion, road capacity and value of time are all part of the first step effects of CAVs to which traffic control applications are closely related. The second step effects include vehicle design and infrastructure while the third step effects contain wider implications such as air pollution, energy consumption etc. When

CAV-enabled traffic control systems are used, it is highlighted that more than two-fold road and intersection capacity benefits can be realised as CAV penetration rate increases in traffic.

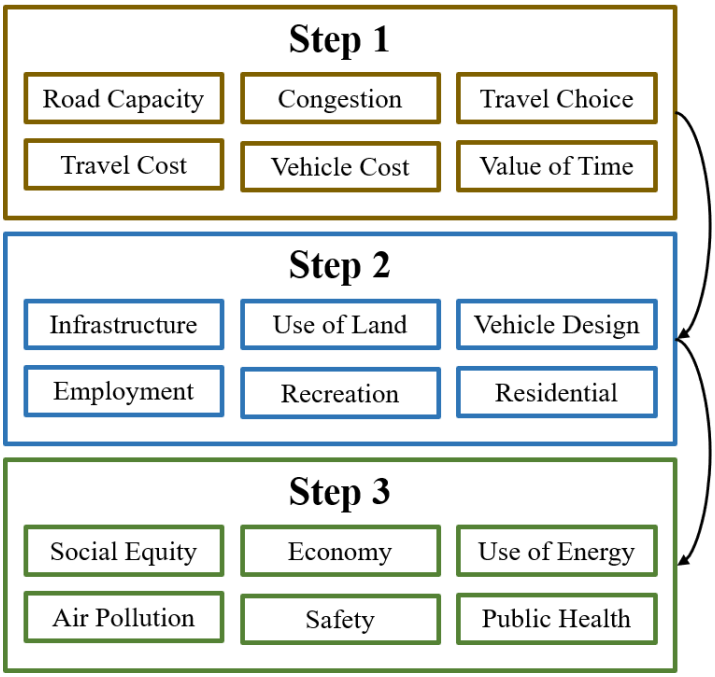


Figure 1 – The sequential effects of autonomous driving from short to long term are shown in 3 steps (Modified from source: Milakis, van Arem and van Wee (2017)).

It has commonly been assumed that removing the human driver element out of the driving task will increase safety and reduce incidents in our transport network (Mladenovic, Abbas and McPherson, 2014) as the great majority of incidents occur due to human error (Fagnant and Kockelman, 2015). Based on the statistics reported by European Commission (2018), obstruction to view, distraction, inadequate plan and insufficient knowledge are among the reasons for incidents caused by human drivers at traffic intersections, and the most common reason is the information failure between drivers and traffic infrastructure, or between driver and vehicle (16% of fatal incidents). This points out a clear need for better information exchange between traffic users at intersections.

The aforementioned advantages of CAVs compared to legacy vehicles are not only due to autonomous features, but connectivity will also play a key role to enable cooperation between vehicles and infrastructure via V2X communications (Fagnant and Kockelman, 2015). Taken together, the literature on this subject supports the notion that many processes and systems will be affected by the advancement of CAV technologies, and the evidence reviewed so far suggests a pertinent role for CAV-enabled traffic infrastructure.

2.1.4. The Impact of CAVs on Traffic Flow

CAV technology is expected to have significant impact on how vehicles operate on the transport network. Speed profiles, platoon formations, the interaction with legacy vehicles, lane positioning and alignment and spacing between vehicles while driving are some of the characteristics that will influence vehicle dynamics, road capacity, safety, carbon emissions and journey times (Atkins, 2016a). 40% penetration rate of CAVs is reported to be a critical threshold to gain significant benefits (>10%) on road capacity, and 100% penetration rate of CAVs could double the road capacity (Milakis, van Arem and van Wee, 2017).

The choice of vehicle behaviour, cautious to assertive, that is implemented by vehicle manufacturers and configurable by the end user can have a great impact on road capacity as the rate of CAV penetration increases (Atkins, 2016a). Cautious driving behaviour, in this case, refers to an autonomous driving style with user comfort (i.e. smaller rates of acceleration or deceleration) and more safety (i.e. larger headways[1] between vehicles than traditional human-driven vehicles) as the main considerations. Considering that human drivers will co-exist in traffic with CAVs for the foreseeable future, it is essential to measure the impacts of different driving behaviours in the presence of CAV-enabled traffic infrastructure at intersections.

Travel time reliability can be improved with CAVs due to advanced vehicle motion control, reduced headways between vehicles and quicker reaction times (Fagnant and Kockelman, 2015). In terms of travel time optimisation with CAVs, intersection control scenarios have more room for improvements compared to highway scenarios (Milakis, van Arem and van Wee, 2017).

2.2. Signalised Traffic Control Systems

There are several TLC methods which will be placed in three broad categories in this literature review: fixed-time, actuated and adaptive control. All of these three control categories make use of the concepts of “phases” and “stages”, and therefore, it is important to understand what they refer to. British Standards Institution (2007) defines the phase as a set of traffic movements that can take place simultaneously, and the stage is defined as part of the control cycle during which a specific set of phases are granted right-of-way with green light. The rest of this section will present the key research work in the aforementioned categories. In addition, traffic control methods for the next generation of vehicles with connectivity and autonomy features are also presented.

[1] Headway is the distance between vehicles on the road measured in time or space.

2.2.1. Fixed-Time Traffic Signal Control

In fixed-time traffic signal control, green phase durations for the approaching lanes and their orders are set offline prior to field deployment with optimisation tools such as TRANSYT (Robertson, 1969), SYNCHRO (Husch and Albeck, 2003) and VISGAOST (Stevanovic et al., 2008). The optimisation is done based on the historical traffic data for a given intersection and there are two main optimisation techniques under the fixed-time control, namely stage-based and phase-based optimisations. The aforementioned tools can generate a series of signal plans prior to deployment in the field for different times of a day or for rush-hour type recurring events.

The fixed-time traffic control does not meet the fluctuating traffic demand and is not capable of responding to any disruptive events such as accidents which is considered as the main drawback (Li, Elefteriadou and Ranka, 2014). On the other hand, it is cost-effective as there is no requirement for installation of sensors and communication devices for real-time data collection.

2.2.2. Vehicle-Actuated Traffic Signal Control

Vehicle actuated methods were introduced to overcome the inefficiencies of offline traffic timing optimisation methods (Gokulan and Srinivasan, 2010). MOVA (Peirce and Webb, 1990), LHOVRA (Kronborg and Davidsson, 1993) and SOS (Kronborg, Davidsson and Edholm, 1997) are prime examples of vehicle actuated methods in which loop detectors play an important role to detect approaching vehicles and set the green time duration accordingly (Abdulhai, Pringle and Karakoulas, 2003). Under this method, the phase time is extended in steps of seconds based on the traffic demand that is detected by the sensors.

Even though the vehicle actuated methods responds to fluctuating traffic demands unlike the fixed-time control methods, it is argued that the control decisions are suboptimal due to their myopic nature in time (Yau et al., 2017; Xie et al., 2012). Another disadvantage of this method is that it only considers the traffic demand on the current phase when deciding whether to increase the green time duration or not for that phase without taking into account the demand on other phases, and this can significantly limit the optimal usage of time and intersection space (Zhao, Dai and Zhang, 2012).

2.2.3. Adaptive Traffic Signal Control

Adaptive control is a combination of the fixed-time and actuated methods in which signal time optimisation is done online at every control cycle i.e. every 15 minutes based on the traffic demand. More precisely, adaptive control includes a traffic network model that takes real-time measurements rather than historical values as input to the model. The split, offset, cycle and/or switching times are the outputs of the model for the subject traffic intersection.

Some of the well-known adaptive control methods are SCATS (Sims and Dobinson, 1980), SCOOT (Hunt et al., 1981), GLIDE (Keong, 1993), ACS-Lite (Luyanda et al., 2003) which are used for urban intersections worldwide (Gokulan and Srinivasan, 2010). Other adaptive control methods include OPAC (Gartner, 1983), PRODYN (Henry, Farges and Tuffal, 1984), RHODES (Head, Mirchandani and Sheppard, 1992), TUC (Dinopoulou, Diakaki and Papageorgiou, 2006), COP (Sen and Head, 1997), DYPIC (Robertson and Bretherton, 1974), ALLONS-D (Porche and Lafortune, 1999), CRONOS (Boillot, Midenet and Pierrelée, 2006) and RT-TRACS (Gartner, Pooran and Andrews, 2002).

Adaptive control can be coordinated for large networks to deal with oversaturation in traffic and the inaccuracies of local sensor measurements, and, it is also considered widely for integration with further traffic control strategies such as freeways (Papageorgiou et al., 2003). Integrated traffic control can maximize the technical and performance benefits of multiple different subsystems by combining different traffic signal parameter update strategies (Wang et al., 2018). On the other hand, achieving real-time performance with adaptive control methods has shown to be very challenging as the model state space grows exponentially with the number of steps in the optimisation horizon (Papageorgiou et al., 2003). Xie et al. (2012) list some of the methods that are implemented such as shorter optimisation horizon, smaller number of phase switches, value approximations and heuristic searches to achieve real-time tractability with adaptive traffic control.

2.2.4. Traffic Signal Control with Vehicular Communications

The traffic signal control methods presented so far summarised the performance improvement efforts made over the years to meet the ever-increasing demand in land traffic without using vehicular communications. The research work presented in this section focuses on advanced real-time traffic control methods that support V2X communications. The core idea is to extract and utilise valuable information in the data that is exchanged between traffic users.

In the absence of vehicular communications, traffic flow parameters are estimated such as flow rate, lane occupancy, average vehicle speed etc. to calculate signal timings. When vehicular communications is used, estimation inaccuracies and challenges are overcome by providing actual real-time data (Li, Wen and Yao, 2014). de Luca et al. (2017) study the traffic signal optimisation problem with vehicular communications under a mixed traffic environment with traditional vehicles and CVs. Stage sequence of the traffic light, the departure times and routes of the approaching CVs are optimised through a meta-heuristic algorithm which is formulated as a Mixed-Integer Program. Similarly, Chang and Park (2013) and Gokulan and Srinivasan (2010) propose signal control strategies that replace estimation functions with real-time data acquired via vehicular communications.

Vehicular communications technology also paved the way for further research work in traffic control that would otherwise not be possible. One prime example is vehicle trajectory optimisation to smooth traffic flow which is commonly referred to as Green Light Optimised Speed Advisory (GLOSA) systems (Katwijk and Gabriel, 2015). The key idea is to exchange signal timings with CVs so that they can apply optimised acceleration and deceleration profiles while approaching an intersection from a certain distance to prevent stop-and-go movements and to reduce energy consumption. Traffic signal timing and vehicle trajectory optimisations together can be considered as a bi-level optimisation problem in which speed profiles are generated based on methods such as model predictive control (Du, HomChaudhuri and Pisu, 2017)(Katwijk and Gabriel, 2015), branch-and-bound (Yang, Guler and Menendez, 2016), heuristic rules (Zhou, Li and Ma, 2015) and rolling horizon (Li, Elefteriadou and Ranka, 2014).

Based on the research work in this field, it becomes apparent that prevalence of vehicular communications also triggered a transition in the design philosophy of traffic control systems. Estimation of current traffic state based on loop detectors is enhanced by accurate real-time data enabled by V2X communications (Chang and Park, 2013).

2.3. Unsignalised Traffic Control Systems

In this research work, unsignalized traffic control is used to define traffic control systems that do not involve traffic lights to indicate right of way. Traditionally, at unsignalized intersections such as roundabouts, T-junctions etc., the driver must make the decision of when to enter the intersection as there is no positive indication or control signal given to the driver by any traffic control infrastructure. The driver waits for a safe opportunity to cross and this driver behaviour is modelled and named as gap acceptance (FHWA, 2001).

Unsignalised intersection control can be categorised as centralised or decentralised based on the decision-making strategy. In centralised intersection control, there is at least one decision made for all vehicles in the control region by a central controller whereas in decentralised intersection control, all decision making is done by vehicles themselves (Rios-Torres and Malikopoulos, 2016). This section will review the literature in intersection control without traffic lights for CHVs and CAVs.

2.3.1. Centralised Control

The vehicle scheduling problem at intersections can be considered as a shared resource allocation in which space and time are discretised and allocated to vehicles. A study published by Naumann, Rasche and Tacken (1998) is the first to propose an alternative centralised traffic control strategy considering the foreseen features of CAVs. The authors proposed a reservation-based intersection control method for vehicles with advanced automated and wireless communication features.

Another seminal study in this area is the work of Dresner and Stone (2004). They also propose a reservation-based intersection control system for CAVs. In this control system, vehicles request and receive time slots via V2I communication interface during which they can traverse the intersection space they reserved. To allow simultaneous access of vehicles with non-conflicting trajectories, the intersection is divided into a n x n grid of tiles. Therefore, each tile can be reserved by only one vehicle per time step based on the First-Come-First-Served (FCFS) scheduling policy. The authors show through simulations that the proposed control system outperforms traditional signal control significantly in terms of average delay experienced by vehicles. Some of the limitations of this system are the inability of vehicles to make a turn and to change their velocity while in the intersection. These limitations are addressed in a follow-up study by Dresner and Stone (2005) and they demonstrate through simulations and field tests (Quinlan et al., 2010) that this augmented system outperforms the signalised control and the stop sign control in terms of average vehicle delays.

Levin, Boyles and Patel (2016) argue that there are certain scenarios in which traffic signal control outperforms reservation-based intersection control with FCFS vehicle scheduling approach. For instance, the fairness objective of a reservation-based control was found to increase the total vehicle delay in an arterial road as side roads were given priority based on their waiting time to cross. Another scenario is that vehicles on the side roads disrupt the platoon progression on the arterial road by obtaining reservations that conflict with the platoon members. It is also reported by Khayatian *et al.*

(2020) that the FCFS method performance degrades as the traffic demand increases which can limit the usage of such a method in the field. However, these limitations can be overcome by implementing a priority-based scheduling with appropriate objective functions.

Taken together, the studies presented thus far support the notion that the FCFS scheduling policy performs worse than the traffic light policy in terms of average delays under heavy traffic conditions. Much of the current literature on scheduling policies for unsignalized traffic control pays particular attention to priority assignment based on Model Predictive Control (MPC) (Camacho and Bordons, 2007) with optimization objectives that include reducing average vehicle delays (Cai et al., 2014; Jin et al., 2012; Zhu et al., 2009) increasing vehicle throughput and shared intersection space utilisation (Fayazi, Vahidi and Luckow, 2017; Altche and de La Fortelle, 2016; Ghaffarian, Fathy and Soryani, 2012), reducing carbon emissions (Lee et al., 2013; Lee and Park, 2012; Huang, Sadek and Zhao, 2012), increasing passenger comfort by considering vehicle dynamics (Dai et al., 2016; Yang et al., 2016), increasing platoon formations (Vial et al., 2016; Tachet et al., 2016; Cheng et al., 2016; Tallapragada and Cortés, 2015; Chen and Kang, 2015; Shahidi, Au and Stone, 2011; Lam and Katupitiya, 2013), and introducing auction and market-based token systems (Carlino, Boyles and Stone, 2013; Vasirani and Ossowski, 2009; Schepperle and Böhm, 2008).

The priority assignment strategy runs in discrete time in the aforementioned studies. The centralised Intersection Control Agent (ICA) gathers the information from each vehicle every control cycle via V2I communication interface. In each cycle, the right-of-way list is generated and communicated back to the vehicles. One of the important outcomes of the studies presented under priority assignment is that finding an optimal crossing sequence of vehicles is more effective in terms of reaching the target objectives than finding an optimal vehicle trajectory for a given sequence of vehicles (Altche and de La Fortelle, 2016). Vehicle trajectory, in this case, refers to the path that a vehicle drives through at an intersection as a function of time. This emphasizes the importance of vehicle sequencing for unsignalized intersection control.

There are a number of studies (Yan, Wu and Dridi, 2014; Gregoire and Frazzoli, 2016; Wu, Abbas-Turki and Moudni, 2009; Ahn and Del Vecchio, 2016) that examine the relationship between scheduling problem in operational research and vehicle sequencing problem at intersections. Vehicles are modelled as jobs whereas the central intersection controller is modelled as a single machine. Yan, Wu and Dridi (2014) extends the single machine job scheduling problem to parallel scheduling for vehicle stream groups that can traverse the intersection simultaneously due to non-conflicting trajectories. Wu, Abbas-Turki and Moudni (2009) argue that a job scheduling algorithm does not meet the real-time operation

requirements that are expected from a safety-critical system as the number of vehicles and lanes increase. For this reason, Ahn and Del Vecchio (2016) convert the job scheduling problem into a Mixed-Integer Linear Programming (MILP) problem to meet real-time requirements. This modified scheduling approach is claimed computationally more efficient as there are significantly smaller number of decision variables.

Hult et al. (2015) investigate the problem of optimal autonomous vehicle control at intersections with safety constraints. The key contribution of their study is the decomposition method that separates vehicle sequencing problem from optimal vehicle trajectory planning. Approaching vehicles are ordered in a centralised way and intersection occupancy time windows are calculated as a first step. The occupancy time windows are then communicated back to the vehicles for the trajectory optimiser that runs locally in vehicles to find optimal and feasible trajectories for crossing. It is claimed that the proposed solution reduces the computational cost on the central controller significantly whilst providing the ability to implement various objective functions on the local trajectory optimiser.

Centralised control strategies have their challenges. Single point of failure, scalability to much larger road networks, processing overhead are some of the major issues that require solution prior to deployment in the field. It is also important to highlight that research on centralised control strategies presented so far has been mostly restricted to all vehicles having a wireless communications interface or all vehicles having fully autonomous features. However, this will not be realised in the near future.

2.3.2. Decentralised Control

In decentralised traffic control, cooperation among vehicles is established through wireless communication without requiring a road side unit acting as the central controller. This is an overlapping area with vehicle autonomy and the integration of these two concepts gives us cooperative and autonomous vehicles (Englund et al., 2016). A preliminary work on decentralised control is studied by (Neuendorf and Bruns, 2004) for platoon of autonomous vehicles. In the concept they introduced, vehicles calculate the right-of-way in a cooperative way as they exchange data among themselves and reach a consensus on when each vehicle should cross an intersection.

Much of the decentralised research has focused on formulating an optimisation problem for sequencing vehicle crossings. Katriniok et al. (2017) adopt an MPC-based approach to generate speed profiles for each vehicle over a finite horizon window such that a cost function is minimised. The cost function is the weighted sum of multiple optimisation parameters including mobility and safety constraints. The

control method does not particularly decide on a crossing order for vehicles but gives a priority relation in the case of trajectory conflicts. Other optimisation methods in the literature focus on minimising the acceleration rate (Zhao, Malikopoulos and Rios-Torres, 2017), calculating the vehicles' degree of freedom to avoid collisions (de Campos, Falcone and Sjoberg, 2013; Hafner et al., 2013), taking the inertia of the vehicles into account to reduce overall energy consumption (Makarem and Gillet, 2011) and minimising the average crossing time through an intersection (Gregoire, Bonnabel and de La Fortelle, 2013).

Game-theoretic approaches are also applied in the literature (Elhenawy et al., 2015; Wu et al., 2016) for vehicle negotiation at intersections in a distributed way. The reward function determines which one of the vehicles with conflicting trajectories need to yield in order to prevent collision. Heuristic rule-based methods are also proposed to solve the distributed control problem (Rodrigues de Campos et al., 2017; Yang and Monterola, 2016; Wu et al., 2015; Hassan and Rakha, 2014). It is also shown via simulation work in these studies that even though a heuristic rule-based strategy offers reduced processing complexity and high scalability features, the solution is sub-optimal (Rodrigues de Campos et al., 2017).

In view of all that has been mentioned so far, decentralised control offers some advantages over centralised control such as no requirements for costly traffic infrastructure deployment with V2I capabilities and no single point-of-failure. On the other hand, Khayatian *et al.* (2020) argue that decentralised control can have higher overheads in wireless communications as vehicles need to broadcast their information much frequently than a centralised control approach, and time synchronisation between vehicles is also considered as a major challenge in the absence of a centralised control unit.

2.4. Artificial Intelligence in Traffic Control

AI based algorithms and models enable machines to learn tasks without being explicitly programmed to do so, and they have demonstrated great potential in complex environments such as strategic long-term decision making problems, process control with set point tracking or regulation, and partially-observable systems in which unmeasurable, and/or unreliable data exists (Nian, Liu and Huang, 2020). The aformentioned points are important advantages over traditional control methods that are presented in Sections 2.2 and 2.3.

Traffic control can be considered as a sequential decision making problem which is too complex to apply simple heuristics or rule-based solutions (Bakker et al., 2010). Therefore, computational intelligence algorithms are widely used for traffic control in the literature including fuzzy logic, neural networks and probabilistic methods. Adaptive signal control systems such as OPAC (Gartner, 1983), PRODYN (Henry, Farges and Tuffal, 1984), RHODES (Head, Mirchandani and Sheppard, 1992) are based on dynamic programming. However, dynamic programming requires the state transition model of the system and in this case, due to the stochastic nature of traffic, it is difficult to obtain a model for traffic control (El-Tantawy and Abdulhai, 2010). Besides, the real-time implementation of a dynamic programming based algorithm is challenging due to the curse of dimensionality phenomenon in the state space (Cai, Wang and Geers, 2010). The curse of dimensionality refers to the issue of data sparsity that occurs as the size and dimension of a state space increases.

RL is one of the main machine learning branches alongside supervised and unsupervised learning. RL methods in the context of traffic control will be presented in this section.

2.4.1. Background on Reinforcement Learning

RL is a goal-directed computational approach that involves a sequential decision process to take actions in a given situation in order to maximise a scalar reward signal. In RL, the dynamics of the environment is not given as a priori to the agent, and therefore, the agent has to interact with the environment to discover which actions give more reward. The background information in this section summarises the key concepts and ideas in RL from the book by Sutton and Barto (2018).

Learning by trial and error, optimal control and temporal difference are the three fundamental subjects that formed RL in the late 1980s. RL differs from supervised learning in that there is no human expert that provides labelled data to take correct actions. Most importantly, the actions taken by the agent influence the environment and the next state of the agent whereas in supervised learning, the environment is not affected by the agent's actions.

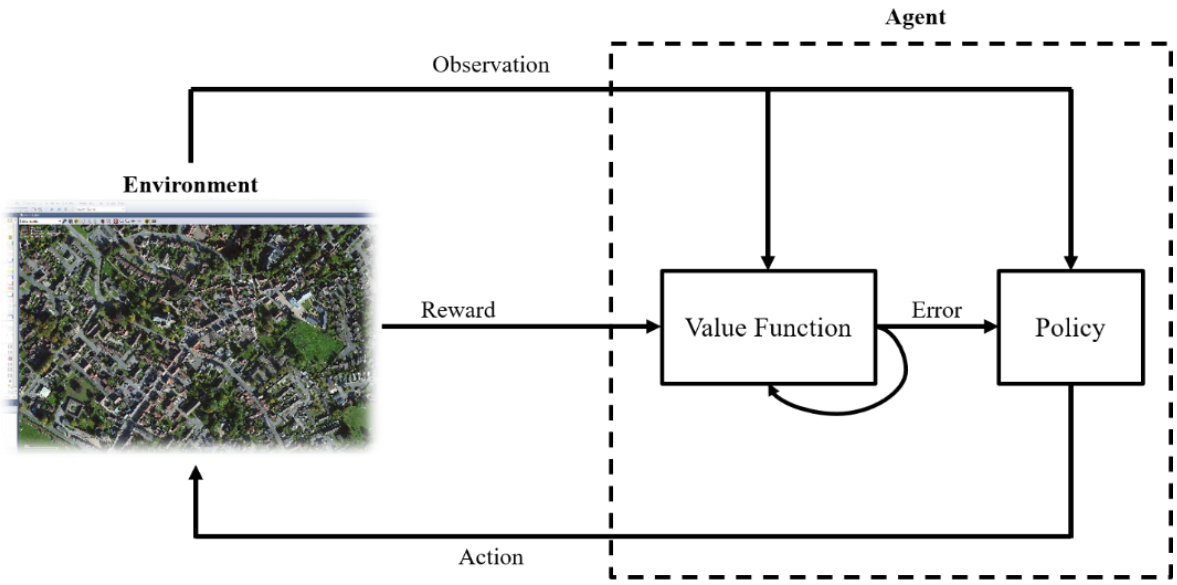


Figure 2 – The interaction between the agent and the environment in RL

RL consists of three main building blocks, a policy, a reward signal and a value function as shown in Figure 2. A policy can be defined as the mapping from states to actions and it forms the behaviour of an agent. A policy can be deterministic or stochastic. A reward signal is used to determine whether an action taken in a certain state gives an immediate good or bad outcome in order for the agent to reach its goal. RL is a goal-oriented learning method in which an agent tries to maximise a scalar reward signal over a period of time. This reward is given by the environment that the agent resides in and is not directly under the agent's control. A value function, on the other hand, is used to define the expected total reward that an agent can receive in the future starting from the current state. Unlike rewards which are given directly by the environment, a value function must be estimated by the agent repeatedly throughout its operation in the environment.

There are several algorithms under the RL umbrella which can be categorised as being either model-based or model-free. The model here refers to a collection of functions that predict the environment dynamics such as state transitions and rewards when a certain action is taken. However, for complex environments, such as traffic, obtaining a representative environment model is challenging (Arel *et al.*, 2010), and therefore, model-free algorithms are predominantly used.

There are two training approaches in model-free methods, namely Policy Optimisation (Silver *et al.*, 2014) and Q-Learning (Mnih *et al.*, 2013). In policy optimisation, the parameters of the agent policy are updated directly while interacting with the environment using the most recent version of the policy.

On the other hand, in Q-learning, the agent policy is not optimised directly but the action-value pair is optimised with data collected at any point during the interaction with the environment. Q-Learning tends to be less stable during training as it indirectly updates the agent policy, however, it has the advantage of being substantially more sample efficient (Szepesvári, 2010). Sample efficiency refers to the amount of training data required to reach a target level in agent performance. With this in mind, there also exists a range of algorithms that can benefit from the advantages of both approaches by concurrently learning an agent policy and an action-value pair. These algorithms are Deep Deterministic Policy Gradient (DDPG) (Lillicrap *et al.*, 2015), Soft Actor-Critic (SAC) (Haarnoja *et al.*, 2018) and Twin-Delayed DDPG (TD3) (Fujimoto, van Hoof and Meger, 2018) which also form the foundations for the proposed unsignalised traffic control method in this research work.

Over the past decade, RL methods in the context of traffic control have been extensively studied by many researchers to improve current TLC methods and to design the next generation systems with CAVs. The next section will summarise the key research work in this area.

2.4.2. Traffic Control Based on RL Methods

RL was first applied to traffic signal control by Thorpe and Anderson (1996). The authors demonstrated through simulation work that RL outperforms the fixed-time traffic signal control in terms of average waiting time in an isolated intersection scenario. Since then, a considerable amount of literature has been published on the application of RL algorithms for traffic control.

There is a large volume of published studies (Abdulhai, Pringle and Karakoulas, 2003; Arel et al., 2010; El-Tantawy and Abdulhai, 2010; Gao et al., 2017; Casas, 2017; Khamis and Gomaa, 2014; Mousavi et al., 2017) describing the role of RL to improve the existing traffic control methods with traditional vehicles. Connectivity and autonomy are not taken into consideration in these studies. The proposed solutions are validated through simulation work and it is argued that accurate traffic state representation improves convergence rate and stability of the RL agent (Van Der Pol and Oliehoek, 2016). The consequence of taking a sub-optimal action in a traffic control application, especially in a busy traffic situation, might lead to a traffic jam which makes returning to a desirable state difficult.

There is also another large body of literature (Gritschneder et al., 2016; Altche, Qian and de La Fortelle, 2016; Ahn et al., 2015; Qian et al., 2014; Perronnet, Abbas-Turki and El Moudni, 2013) that is concerned with RL based traffic control for CAVs. Considering that there will be a long transitional period in which traditional human-driven vehicles and fully autonomous vehicles will co-exist in traffic

(Liu, Ma and Kumar, 2015), it is crucial to accommodate human-driven vehicles while creating traffic control methods (Dresner and Stone, 2006).

2.5. Research Gaps

The main research gaps based on the literature review presented in this chapter can be summarised as below:

- Mladenovic et al. (2016) draw our attention to the technology path dependency for traffic control methods, and it is argued that the legacy traffic control methods should not be the foundation when considering a next generation disruptive technology such as unsignalised traffic control with CAVs. There has been a limited amount of work reported on AI-based unsignalised intersection control and there is research gap to address the challenges of how an AI algorithm should be formulated, trained and validated.

- Much uncertainty still exists about the impact of CAVs on unsignalised intersections in mixed-fleet traffic scenarios where CAVs and CHVs co-exist. The interactions between CAVs and human-driven vehicles at intersections also remain unclear (Atkins, 2016a). Vehicle delays at road intersections are mainly caused by conflicting turning manoeuvres (Fagnant and Kockelman, 2015) which is difficult to address by distributed traffic control strategies. Therefore, a centralised traffic control method plays an important role in this case.

- Recent trends in vehicle automation and connectivity have led to a proliferation of studies in traffic control strategies for the next generation of land vehicles. However, vehicle automation is still in its infancy, and therefore, there is little empirical data from field tests. The state-of-the-art studies mainly used simulation tools to explore the effects of CAVs in traffic and to validate methods and algorithms (Milakis, van Arem and van Wee, 2017). Even though the experimental data gathered from simulation works is based on various assumptions, there is a general agreement that CAVs will increase the efficiency of the transportation system. This indicates a need for practical experiments to investigate the impacts of CAVs in traffic flow at intersections when different traffic control methods are used.

This research work will address the gaps identified and propose an unsignalised intersection control method for CAVs which is based on the RL branch of machine learning. RL can learn the non-linear relationship between the elements that play a role in the intersection control (Arel et al., 2010), from which the agent can derive a control policy for traffic scheduling.

2.6. Goal and Objectives

The primary goal of the research work is the proposal of an unsignalised traffic control method based on the RL branch of machine learning to advance the current state-of-the-art by considering the future of land traffic where CAVs and C-ITS communication systems are prevalent. The purpose is to make real-time control decisions in a proactive way under constantly evolving traffic conditions whilst ensuring a higher degree of safety in the absence of any physical traffic light system. The objectives of the project are as follows:

O1. Identify research gaps during literature review in terms of traffic intersection control methods, wireless communication solutions for ITS applications, CAV technologies and RL strategies that are applicable for stochastic environments such as traffic,

O2. Propose a Vehicle-to-Infrastructure (V2I) wireless communication protocol between Connected Vehicles (CV) and the Road Side Unit (RSU) acting as the intersection control agent, and to identify a data set for the traffic control application,

O3. Design and implement a neural network architecture, traffic state representation, action space and reward mechanism, in the context of RL, to learn an optimal traffic control policy whilst preventing any vehicle trajectory conflicts for collision avoidance,

O4. Create a realistic traffic environment in simulation to validate the proposed unsignalised traffic control method under various scenarios, and benchmark against identified traffic control methods for performance comparison,

O5. Develop a scaled testbed for practical trials in a controlled environment with multiple scaled CAVs and the digital twin of the scaled testbed in the simulation environment to further validate the proposed traffic control method in a cost-effective way,

O6. Analyse the results of simulation work and scaled testbed experiments, and highlight the key findings and the impacts of the proposed control method on traffic flow, congestion and environment,

O7. Draw conclusions on the research work and provide recommendations on potential real-world deployment of the proposed traffic control method considering the challenges in terms of technical, commercial and policy.

Chapter 3

3. Development of Vehicular Communications Protocol for Intersection Crossing

3.1. Introduction

The problem of vehicle coordination at an unsignalised intersection with multiple approaching and exit links and multiple lanes on each link is considered in this work as shown in Figure 3. Permitted left, right turn and through movements are shown with road markings. The Intersection Critical Area (CrA) is defined as the intersection area of incoming lanes $l_1..l_n \ n \in N$ where *N* is the total number of incoming lanes. CrA has the potential for lateral vehicle collision and it consists of Conflict Points (CP). Essentially, only a single vehicle must occupy a CP at any moment in time. With this in mind, unsignalised intersection control problem turns into a spatio-temporal control in which time windows are allocated for each vehicle at CPs to enable safe crossing.

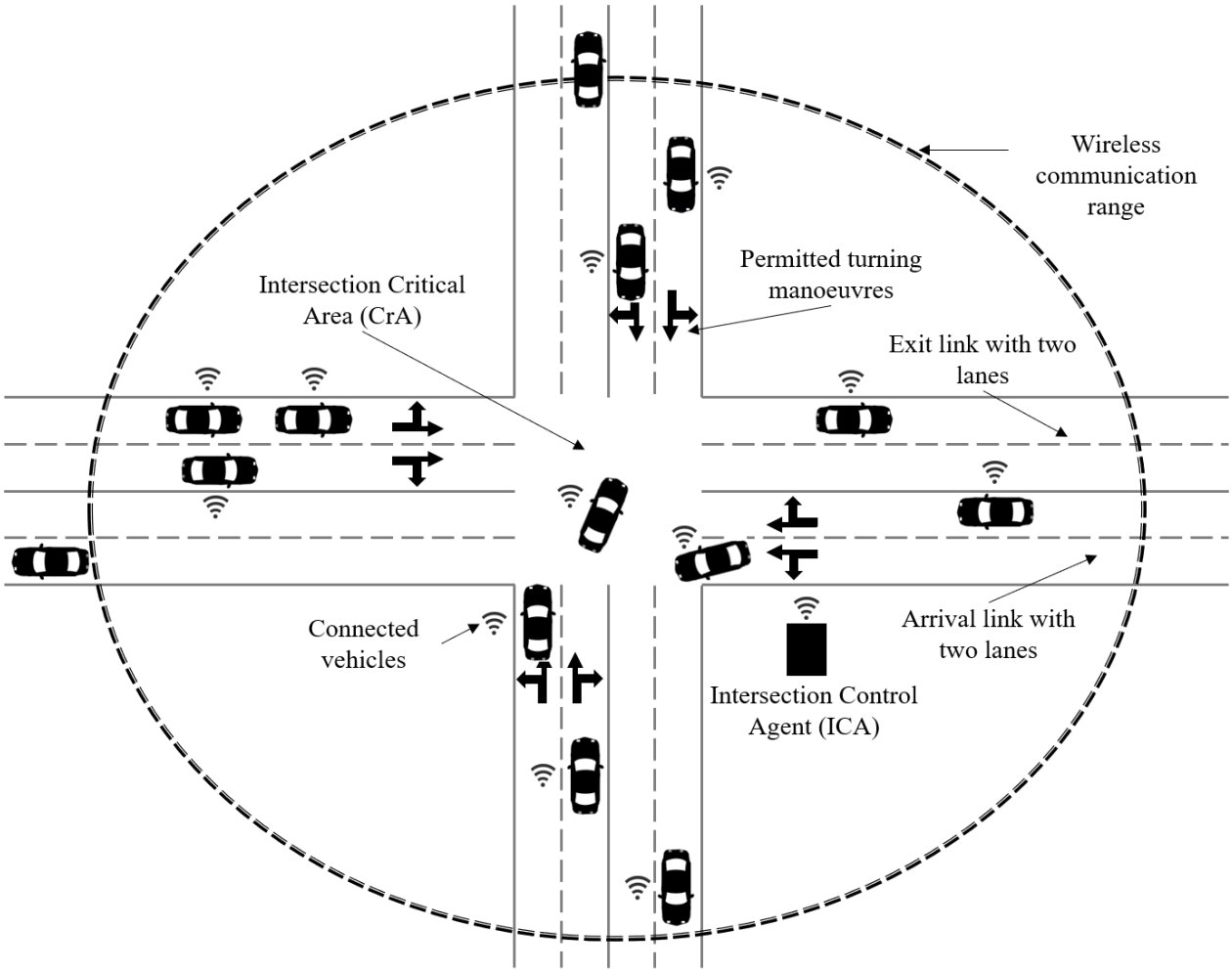


Figure 3 – An example traffic intersection setup and visualisation of key definitions

In a centralised intersection control system, the RSU with V2I communication capabilities collects vehicular data from approaching CVs, and disseminates intersection crossing time windows for each and every approaching vehicle. CVs transmit data such as vehicle identification number, position, velocity, desired trajectory through the intersection etc. when they enter the communication area of the local ICA.

The rest of this chapter gives more detailed information about the wireless communications protocol developed for the intersection crossing of vehicles, data requirements for the unsignalised intersection control and the role of the key components in the control system.

3.2. Assumptions

In this research work, the following set of assumptions are made in order to focus on the proposed contributions. Firstly, vehicles in traffic are assumed to be a mixture of CHVs and CAVs. Traditional vehicles with no connectivity are not considered. Secondly, human drivers have access to an on-board display in their vehicles where appropriate intersection crossing information is displayed, and they obey the allocated crossing time window. Thirdly, all vehicles are equipped with Global Positioning System (GPS) sensors with measurement accuracy of no worse than 1 metre. Finally, Vulnerable Road Users (VRU) such as pedestrians and cyclists are not considered.

Vehicle interactions and driving behaviour outside the intersection communication range is beyond the scope of this research work. The underlying wireless technology and communication concepts for the rest of this chapter is based on the C-ITS standards as explained in Section 2.1.1.

3.3. V2I Communication Model

The communication model for the unsignalised intersection control is shown in Figure 4 as a Unified Modelling Language (UML) deployment diagram and it is a simplified version of the work by Gáspár, Szalay and Aradi (2014) that highlights the essential components for this research work.

The model in Figure 4 demonstrates the holistic view of traffic control and it is designed to handle data for spatially and temporally dynamic traffic environment. In this view of the system architecture, the Local Traffic Control System consists of an RSU, wireless communications radio equipment and supporting sensors that can include video detection, temperature sensors etc. and this research work fits in this component where proposed algorithms and methods are implemented locally in the RSU.

It is also possible that the RSU may not be required at every intersection and that some of the traffic control functions could be supported remotely (Gáspár, Szalay and Aradi, 2014) by a Remote Traffic Management Centre as shown in the far left box in Figure 4 that essentially has regional transportation management responsibilities. Another key component in the communication model is the Vehicle OBU that can be a part of either CAV or CHV to provide secure and reliable wireless communication capabilities. In alternative system designs, some of these responsibilities can be assigned to different components. In the rest of the chapter, the interactions between the ICA and CVs will be detailed and their roles in the unsignalised traffic control will be explored.

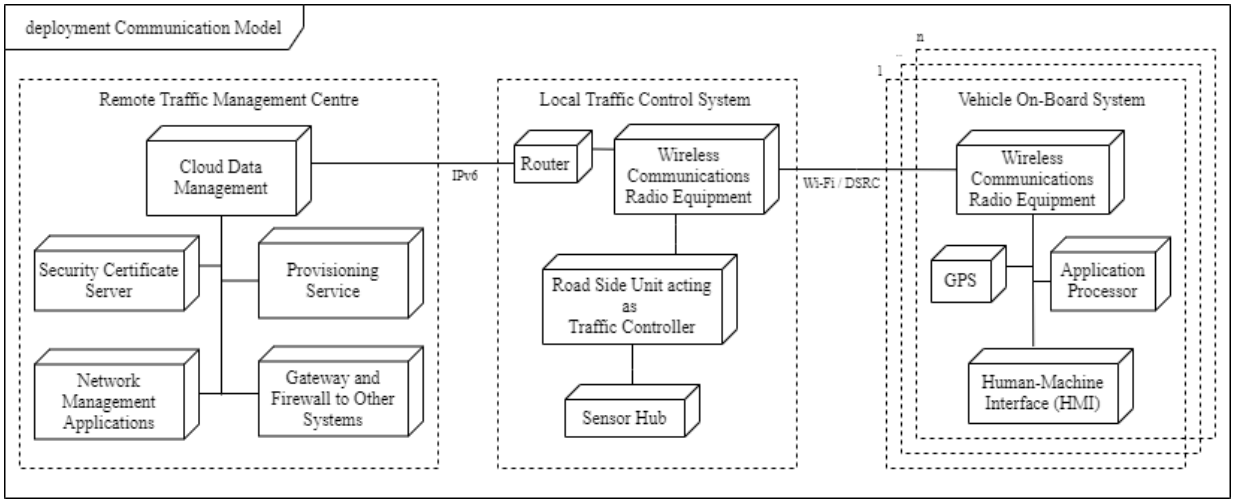


Figure 4 – Communications model UML deployment diagram
(Modified from source: Gáspár, Szalay and Aradi (2014))

3.3.1. The Area of Interest

ICA is responsible for broadcasting regular messages to announce the presence of the unsignalised traffic control service. CVs that are approaching an intersection monitor such messages so that they can participate in the traffic control service, by exchanging data on the designated radio channels (ETSI, 2010b). Therefore, the Area of Interest (AoI) for the unsignalised traffic control system can be defined as the area that CVs can establish direct wireless communication with the RSU.

The AoI is further divided into three areas; Control Area (CoA), Critical Area (CrA) and Exit Area (ExA) in which CVs and the ICA have to execute certain control and monitoring tasks. Figure 5 demonstrates these key tasks in a spatio-temporal way in which the x-axis is the time and the y-axis is the distance from the intersection where d_0 is the intersection entry point. The tasks marked as "V" are executed by OBUs of each Vehicle Agent (VA) whereas the ICA executes the tasks marked as "I". The CVs also transition into different modes as "Approaching", "Crossing" and "Exiting" within the AoI as shown on the left side of the diagram.

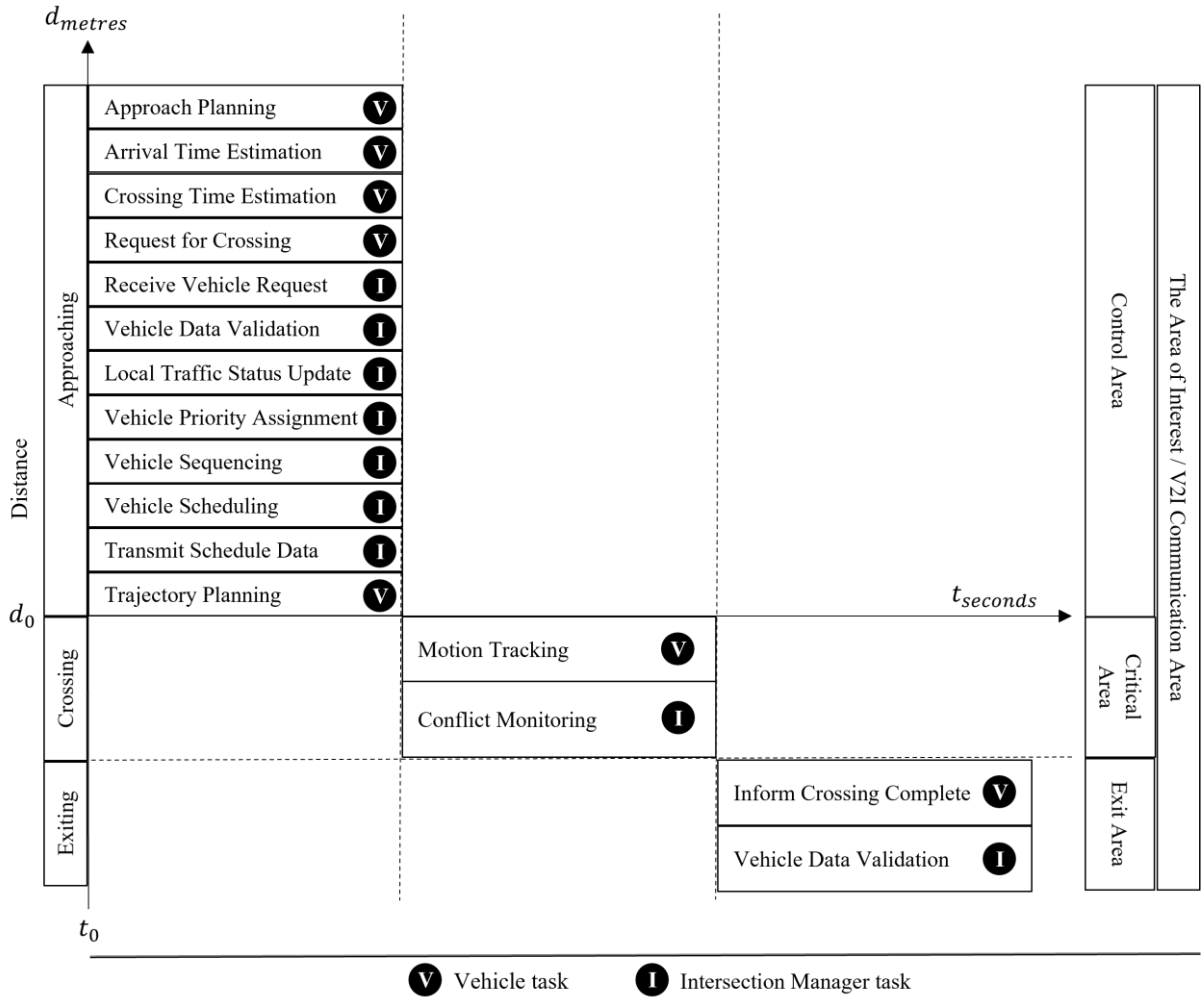


Figure 5 – The Interaction diagram that demonstrates the key tasks of VAs and the ICA in a spatio-temporal way

When a CV enters the AoI, the first task it executes is the Approach Planning which includes mapping its own geo-location on the intersection map data received, positioning itself on the correct lane based on its desired trajectory through the intersection and the turning manoeuvre restrictions within the CrA. The map data is based on the common data dictionary by ETSI (2014b). Once a vehicle is positioned in the correct lane, it calculates an Estimated Time of Arrival (ETA) to the intersection entry point with free flow traffic assumption as in Highway Capacity Manual by Transportation Research Board (2010) and given by the equation:

$$t_{ETA} = \frac{d_{int}}{v_{tar}} + t_{init} \quad (1)$$

where v_{tar} is the target speed of the vehicle which depends on its driving behaviour and the speed limit in the local road network and d_{int} is the distance of the subject vehicle to the intersection entry point once target speed is reached. t_{init} is the initial acceleration or deceleration time that it takes for the subject vehicle to reach the target speed starting from its current speed v_{curr} and is calculated as below:

$$t_{init} = \frac{v_{tar} - v_{curr}}{a_d} \tag{2}$$

where a_d is the desired acceleration or deceleration value. Finally, the vehicle displacement d_{init} during t_{init} period is calculated by the equation:

$$d_{init} = \frac{{v_{tar}}^2 - {v_{curr}}^2}{2a_d} \tag{3}$$

from which, the d_{int} can simply be found as:

$$d_{int} = d_{curr} - d_{init} \tag{4}$$

where d_{curr} is the current distance of the vehicle from the intersection entry point. It is important to highlight that desired vehicle driving behaviour in terms of target speed, acceleration etc. is decided by the vehicle itself and respected by the ICA rather than imposing on speed profiles determined by the ICA for all approaching vehicles.

The Crossing Time Estimation task calculates the time that it takes for the subject vehicle to cross the CrA depending on the target manoeuvre i.e. left turn, straight etc. and the target speed while making manoeuvre. Once the subject vehicle executes the aforementioned set of tasks, it transmits this information to the ICA for intersection crossing request. The data requirements within the context of unsignalised intersection control are detailed in the following section.

3.3.2. Vehicle Agent Data

When a CV is within the AoI and executes the initial tasks that are explained in the previous section, it is then responsible for transmitting its data to the ICA for traffic control actions to be taken. The vehicle data includes information such as vehicle dynamic state, driving behaviour, vehicle configuration, communication parameters and trajectory path through the intersection. ICA receives this information from all vehicles in the AoI, executes data authentication checks with the remote TMC, and captures the real-time local traffic state.

SAE International (2020) specifies a V2X communications message set dictionary that is agnostic from the underlying communication technology i.e. ITS-G5, IEEE 802.11, 4G etc., and it is named as J2735. This dictionary is intended specifically for applications that use V2X communications systems, and it describes a message set and its data frames. Table 1 lists a key sub-set of this dataset which is used as part of the proposed unsignalised traffic control method between the vehicles and the ICA.

VA Data	Description
veh_id	Unique vehicle identification number
msg_id	Message identification number
msg_t_tx	Transmit message timestamp
ica_id	ICA identification number
veh_type	Vehicle type including the autonomy level
dist_to_cra	Distance to the intersection CrA
veh_v	Current vehicle speed
veh_a	Current vehicle acceleration / deceleration
veh_length	Vehicle length
veh_width	Vehicle width
veh_t_arr	Estimated arrival time to the critical area
veh_lane_arr	Intersection arrival lane
veh_lane_exit	Intersection exit lane
veh_v_cross	Vehicle target crossing speed
veh_dt_cross	Time to cross the entire CrA at veh_v_cross

Table 1 – Vehicle dataset transmitted by all CVs within the Area of Interest when Unsignalised Traffic Control service is available.

3.3.3. Intersection Control Agent Data

The ICA processes the received data from the CVs every control cycle to generate vehicle priorities and schedule their crossing time windows. The control cycle term refers to the iterative process of buffering approaching vehicle information, assigning priorities for each buffered vehicle and sending crossing time windows to those vehicles. From the ICA point of view, buffering more vehicles by extending the duration of the control cycle helps towards making more optimal decisions as solution space for giving priority to vehicles increases. However, the trade-off is that a less frequent crossing time window allocation may cause vehicles to reach at the intersection entry point and stop due to having no valid crossing time window yet. Therefore, in this research work, the control cycle duration is implemented as 5 seconds based on the experiments conducted in the simulation tool.

The key data transmitted by the ICA to all vehicles is summarised in Table 2 and this includes the details of the addressed vehicle, the assigned vehicle priority relative to other vehicles in the local network, the allocated time window to cross the intersection and any deviations from the proposed

request to cross by the vehicle itself such as reduced crossing speed or exit lane etc. No vehicle is allowed to cross the intersection if no crossing time window slot is allocated by the time a vehicle reaches to the intersection entry point. This can happen due to multiple reasons including wireless communication issues, traffic jam in the exit link, traffic incident in the intersection CrA etc. Therefore, It can be considered as a safety feature to prevent crashes.

ICA Data	Description
veh_id	Unique vehicle identification number
msg_id	Message identification number
msg_t_tx	Transmit message timestamp
ica_id	ICA identification number
veh_t_cross	Crossing window start time
veh_prio	Allocated vehicle priority relative to other vehicles

Table 2 – Intersection control dataset transmitted by the RSU for each CV within the Area of Interest when Unsignalised Traffic Control service is available.

3.4. V2I Communication Protocol

Unsignalised traffic control system can be considered as a point-to-multipoint communication system as specified in ETSI (2010) from the ICA point of view. The data between the ICA and CVs is exchanged from the originating source in a single hop to the receiving node located in the direct communication range by using the control channel G5-CCH of the C-ITS communication architecture.

The ICA is responsible for sending two types of messages: CAM and DENM (ETSI, 2010b). A CAM message includes the presence of intersection control service, detailed topological and geometrical information about the intersection and information related to intersection crossing for each vehicle. DENM is an event-based message, and it is used to report any road hazards or abnormal traffic conditions in the local road network. The maximum communication latency for both cases must be 100ms (ETSI, 2009).

3.4.1. Intersection Crossing Request

CVs in the AoI subscribe to the unsignalised traffic control service by sending a “Request for Crossing” message. This message essentially registers the vehicle details in the ICA for spatio-temporal crossing window allocation by the ICA. After the request is sent, the vehicle receives an acknowledgement message back from the ICA as shown in the communication diagram in Figure 6.

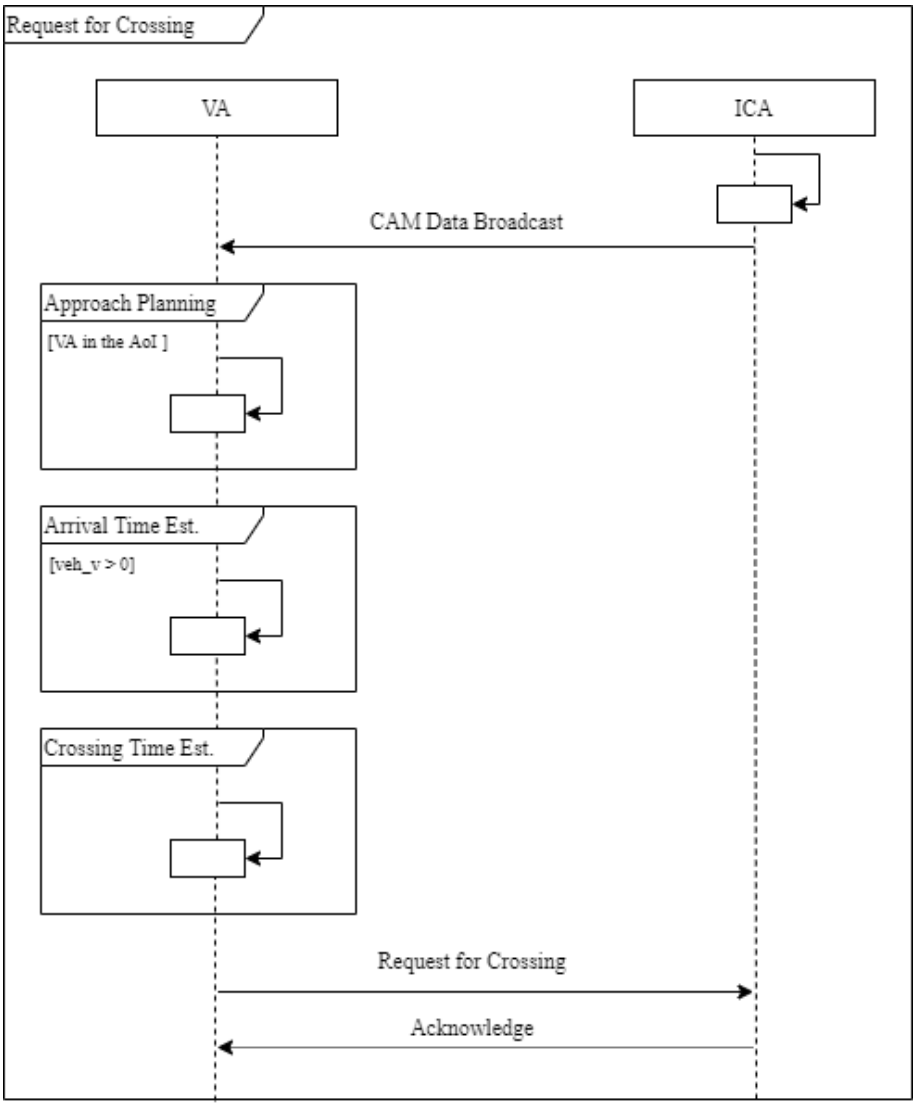


Figure 6 – Request for crossing communication diagram.

A timestamp of the transmitted data is one of the most important parameters for the ICA to realize the unsignalised traffic control application as the age of highly dynamic data affects safety. The ICA might take incorrect actions if the transmitted data does not correspond to the latest state of the vehicle i.e. trajectory, speed etc (ETSI, 2013). For that reason, the safety and efficiency of the unsignalised traffic control service depends on the timely delivery of the transmitted data from the approaching vehicles.

When the “Request for Crossing” message is sent, the vehicle expects a crossing time window from the ICA which may start at the estimated arrival time of the vehicle to CrA entry point. This is the ideal condition in which the stop-and-go movement of the vehicle is prevented (Katwijk and Gabriel, 2015). However, when traffic is congested, the majority of the approaching vehicles will have to wait at the CrA entry point until their allocated crossing time window starts which is also the assumption that vehicles make until they receive a valid crossing time window on their approach to the CrA entry point. This assumption ensures safety as vehicles stop at the CrA entry.

The ICA buffers the received vehicle data until the next control cycle, and it processes the vehicles in batches starting from the Vehicle Data Validation task as shown in Figure 7. In this stage, cybersecurity management is done according to the security layer processes in ETSI (2010). This ensures that vehicles that are participating in the unsignalised traffic control service are genuine and their data is validated. For the sake of simplicity and focusing on the communication between the ICA and the vehicles, Figure 7 does not include any communication details between the ICA and the Remote TMC for the security certificate management and authentication of the vehicles. Following this stage, the ICA updates its perception of the local traffic state in terms of vehicles that: are in the AoI, require crossing time windows, exit the AoI and report anomaly.

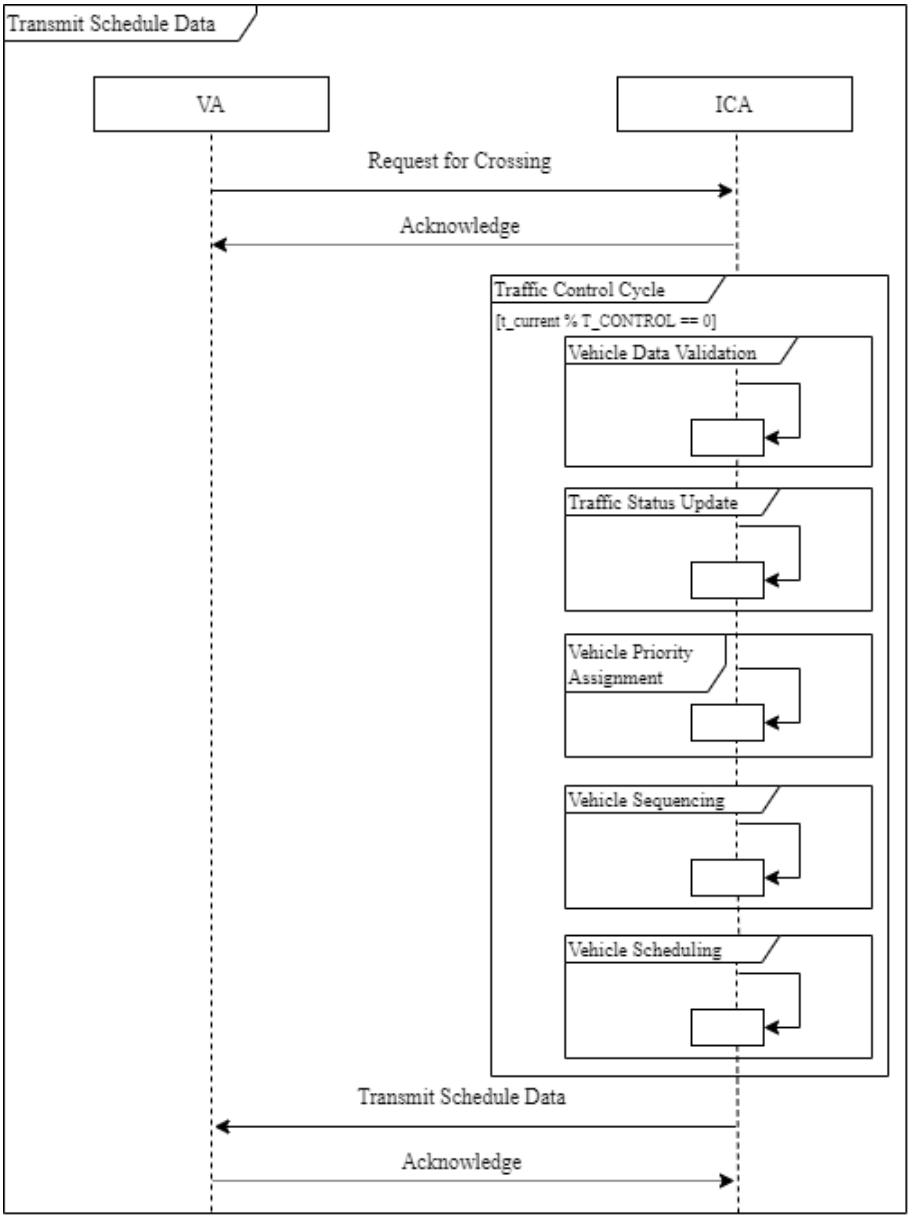


Figure 7 – Transmit schedule data communication diagram.

Vehicle priority assignment, sequencing and scheduling are the tasks which will be explained in detail in Chapter 4. In short, an unordered list of approaching vehicles is ordered and given priorities based on the intersection control objective and crossing time windows are allocated to these vehicles by applying conflict resolution techniques. The output of these stages is the crossing time window data to be transmitted back to the vehicles as a response to their “Request for Crossing” message.

3.4.2. Crossing Time Change Request

When approaching vehicles transmit their unsignalised traffic control service subscription message and obtain a crossing time window from the ICA, their motion tracking algorithm on-board ensures that vehicles arrive no later than the allocated crossing start time. However, if an approaching vehicle is late for the start time due to any problem or change of vehicle trajectory such as selecting different entry or exit lane, they must request for an updated crossing time window as shown in Figure 8.

The ICA processes the request for an updated crossing time window in the next control cycle. Firstly, the previously allocated time window is cancelled, and then, a new time slot is generated in the exact same way as before when “Request for Crossing” message was transmitted. It is important to note that the new crossing time window might affect other vehicles that already hold valid crossing time windows. A special procedure within the Conflict Resolution algorithm (See Section 4.4) is executed to identify the vehicles with conflicting trajectories. These vehicles are also allocated new crossing time windows which are no sooner than their initial crossing time windows and they are notified with a “Crossing Time Update Notification” message.

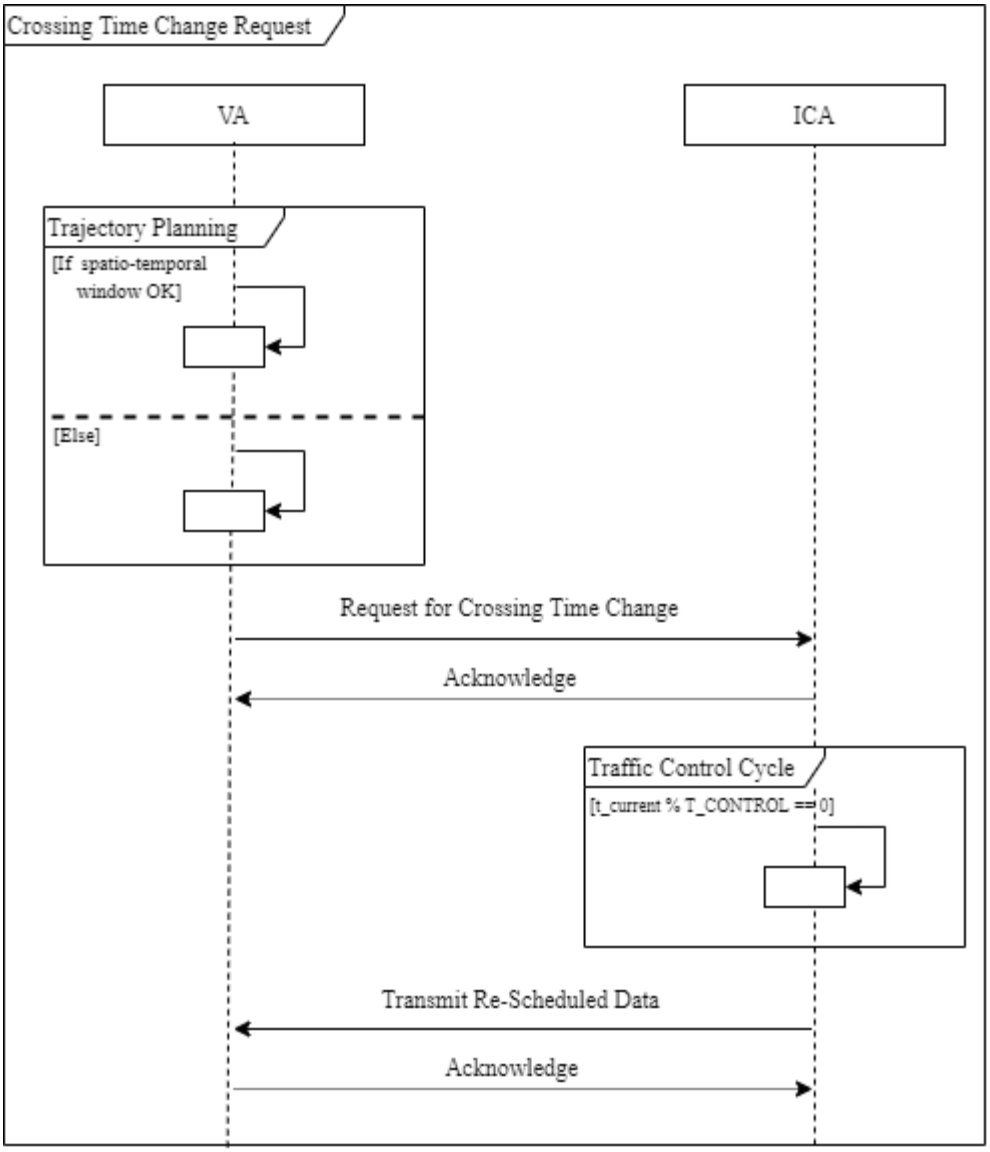


Figure 8 – Crossing time change request communication diagram.

3.4.3. Intersection Exit Notification

The final stage of the unsignalised traffic control service is the exit notification transmitted by the CVs when they are in the Exit Area of the AoI as shown in Figure 9. The ICA de-registers these vehicles from the traffic control service and no longer tracks their status.

In the case of a vehicle malfunction or a traffic incident, a DENM message is transmitted according to ETSI (2014) so that any necessary safety actions can be taken including cancelling all allocated time windows and stopping traffic until the emergency situation is handled appropriately. This type of emergency handling methods is outside the scope of this research work.

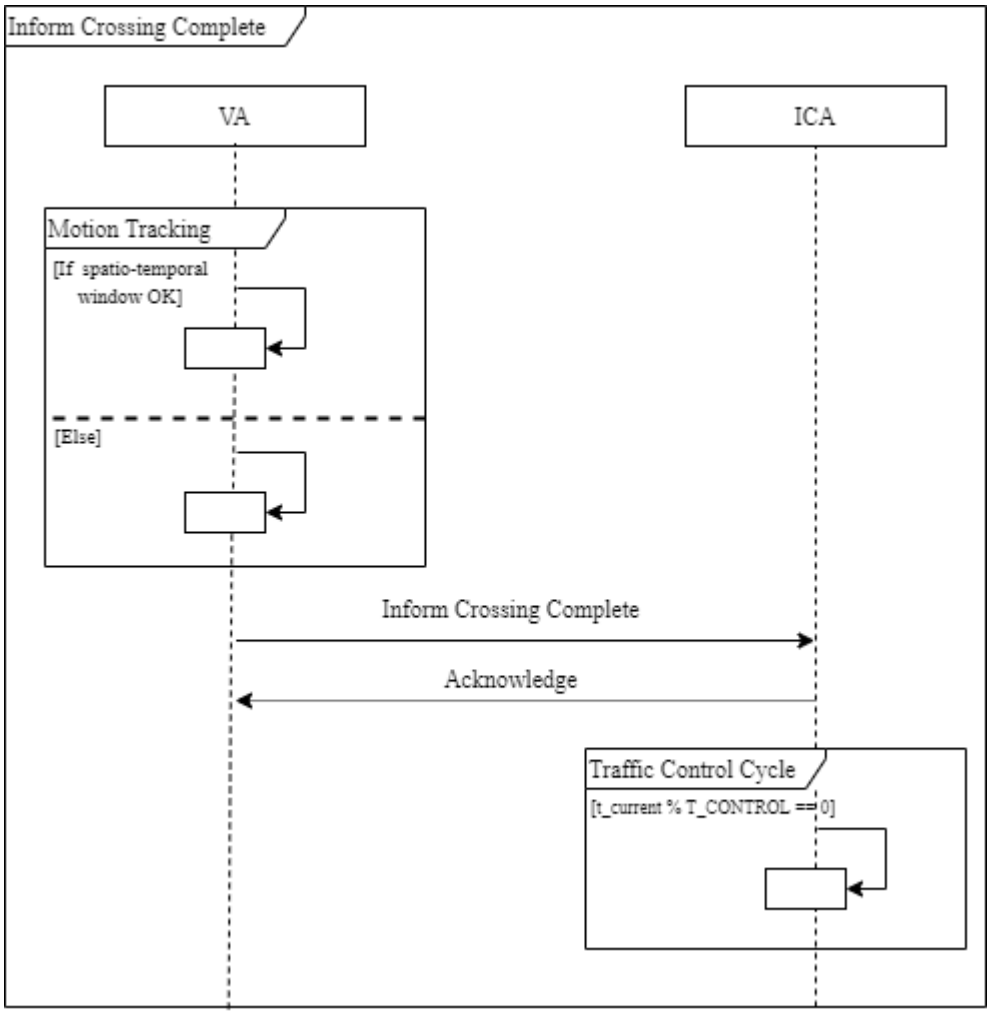


Figure 9 – Inform intersection crossing complete in the Exit Area communication diagram.

3.5. Summary

This chapter presents the unsignalised traffic control problem statement and the V2I wireless system details including the communication protocol and the data requirements. There are certain assumptions taken into account and listed in this chapter for the development of the communication strategy in order to focus on the main contributions of this research work.

Unsignalised traffic control is one of the ITS applications that target at reducing congestion and the risk of collision. Therefore, the communication latency between the CVs and the ICA must not be greater than 100 ms as specified in SAE J2735 even though the performance may vary according to wireless network characteristics, load and radio obstacles. CAM and DENM transmission concepts are used for bi-directional data exchange between the users of the traffic control system based on ETSI communication architecture for V2X applications.

The V2I communication architecture presented in this chapter is based on comprehensive research work and standardisation activities in Europe and the rest of the world. Therefore, the feasibility and compatibility of the proposed system with the existing ITS services and products are targeted. It is important to highlight that the unsignalised traffic control service and the algorithms that will be presented in the following chapters do not depend on a particular communication technology used such as Wi-Fi, cellular, ITS-G5 etc. and it is agnostic from the V2I communication layer. This is primarily to make sure the proposed traffic control system can be deployed at different locations (i.e. urban cities, small villages, near tall buildings etc.) where certain communication technology may perform better than others.

Chapter 4

4. Algorithm Design: AI Traffic Control for Unsignalised Intersections

4.1. Introduction

RL mainly originates from the optimal control and dynamic programming research fields (Nian, Liu and Huang, 2020) in the early 1980s. RL is formulated as a sequential decision making algorithm for problem domains which require consideration for randomness within the system and are too complex to apply simple heuristics or rule-based solution like traffic control (Bakker et al., 2010). Even though the stochastic elements within an environment cannot be controlled, an agent can learn to optimise its actions in the presence of stochasticity.

The RL paradigm in the context of traffic control is shown in Figure 10. There are two main components; the TCA and the traffic environment in which the TCA operates. The TCA receives observation and reward values from the environment, and it makes a decision on the next action based on its policy.

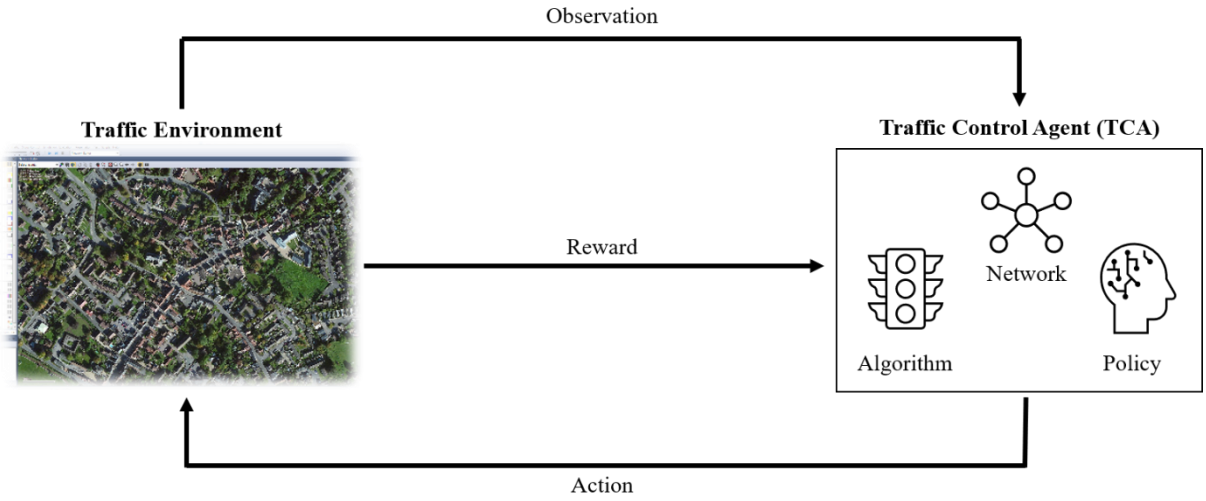


Figure 10 – Environment and agent interactions in RL

In the rest of this chapter, the details of how the TCA probabilistically maps the observations to actions in order to maximise the rewards that it receives are explained. The neural network processes the traffic observations in order to establish the aforementioned mapping and to determine the priority of the approaching vehicles so that the intersection crossing sequence can be generated.

The conflict resolution of the shared intersection space is the next stage in the traffic control cycle once a list of vehicles with priorities are generated. The methodology that ensures vehicles cross the intersection safely at their allocated crossing time windows will be explained.

4.2. Assumptions

The following set of assumptions have been made in order to focus on the proposed contributions and innovation in this section. Firstly, the traffic control system has sufficient computational resource and constant power source to process all required data and to execute algorithms detailed in this section. Secondly, the intersection crossing of all vehicles through the AoI is controlled by the proposed traffic control system that disseminates space-time crossing information.

4.3. Neural Network Model

4.3.1. State Representation

A state in RL gives a complete description of an environment whereas an observation gives a partial or limited description of an environment. A chess game can be given as an example for an environment from which a state can be obtained with no missing or hidden information (Sutton and Barto, 2018). In real world applications, identifying a state is very challenging and generally not possible to obtain, and instead, observations are used. Traffic control is an environment which is too complex to obtain a state and the sheer scale of the control problem requires a carefully constructed observation vector among a plethora of available information. The “curse of dimensionality” is a term used in RL that refers to the exponential growth in computational resource requirements with the number of observation vector variables (Sutton and Barto, 2018). In this research work, an observation vector is constructed that captures the current traffic flow and vehicle states as comprehensive as possible for the TCA to make control decisions while keeping the number of variables in the vector to a minimum.

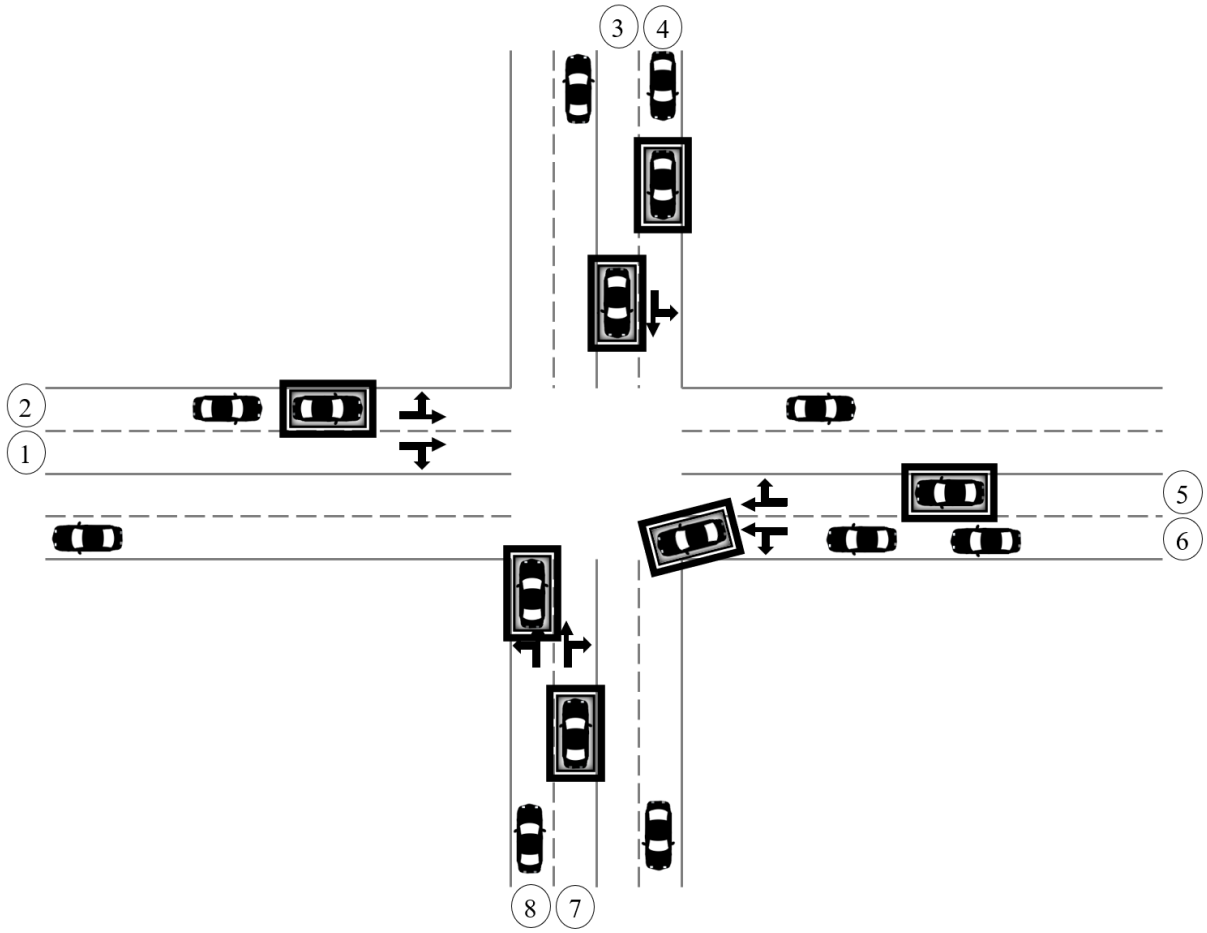


Figure 11 – The representation of the traffic flow and the vehicle states which is updated every control interval.

Figure 11 demonstrates an example of a 4-way intersection with two lanes on each approaching link which are numbered sequentially clockwise direction starting from the west approaching link. The lead vehicle is defined as the vehicle with the shortest distance to CrA on a particular lane and these vehicles are highlighted with black frames in Figure 11. Given that, s_t denotes the observation vector at time step *t* for the TCA and it is given as:

$$s_t = \{ s_1,\ s_2,\ \ \ldots,\ s_N \} \tag{5}$$

Where *N* is the total number of approaching lanes for an intersection and s_t is an aggregation of the observation vectors on each approaching lane. Following that, s_n denotes the observation on a particular approaching lane of an intersection where $n \in \mathbb{N}$ and it represents the lane number. s_n is defined as:

$$s_n = \{ al_{veh}, d_{int},\ t_{arr},\ l_{arr},\ l_{exit},\ v_{lane},\ t_{delay},\ r_{delay} \} \tag{6}$$

where:

- al_{veh}: Autonomy level of the lead vehicle i.e. SAE Level 0, 5 etc. Each autonomy level is represented with a unique identification number.
- d_{int}: Distance of the lead vehicle to CrA in meters on lane *n*.
- t_{arr}: Arrival time of the lead vehicle to CrA in seconds on lane *n*.
- l_{arr}: Arrival lane ID of the lead vehicle.
- l_{exit}: Exit lane ID of the lead vehicle.
- v_{lane}: Average vehicle speed in km/h on lane *n*. This is the weighted average of speed of all vehicles on lane *n*. The weight is the respective travel time of the vehicles. It means that vehicles that have just entered the network have less influence on the value of this calculation than vehicles that have been travelling on the approaching lane longer time.
- t_{delay}: Average vehicle delay in seconds on lane *n*. Average vehicle delay is calculated by dividing the total vehicle delay on lane *n* to the number of vehicles on lane *n*, and the total vehicle delay is the aggregation of the delay values per vehicle which is obtained by dividing the actual distance travelled in the current timestep to the difference of desired vehicle speed and actual vehicle speed.
- r_{delay}: Ratio of the average vehicle delay on lane *n* to the total vehicle delay on all approaching lanes.

The observation vector variables are normalized with respect to a pre-determined maximum value such that all values are in the range of [0, 1]. For example, if the average vehicle speed is 15 km/h and the speed limit is 30 km/h on that lane, then v_{lane} will have the value of 0.5. The scale and distribution of the observation vector values differ from each other, and larger weight values are required in the neural network as the spread of a vector value gets larger (i.e. thousands of units as opposed to tens of units). Therefore, it is common in deep learning applications to apply linear transformations to an input vector before it is fed to a neural network (Bishop, 1995) in order to prevent unstable behaviour, poor performance during model training and high generalisation error during model evaluation.

The state representation that is introduced above essentially consists of the lead vehicle (al_{veh}, d_{int}, t_{arr}, l_{arr}, l_{exit}) and the average traffic flow parameters (v_{lane}, t_{delay}, r_{delay}) for each lane on the approaching links. This kind of representation reduces the observation vector size significantly as opposed to representing the parameters of all vehicles in the observation vector.

4.3.2. Action Space

The set of actions in an environment that an agent can take to reach its goal is called the action space in an RL framework. There are two types of actions, discrete and continuous. A chess game is an example environment with a discrete action space as there are a finite set of available moves for an agent to take whereas a throttle and steering control environment for a CAV has continuous action space where the actions are real number within certain limits. The action space A for the TCA is continuous, and it contains the vehicle priorities for all approaching lanes as shown below:

$$A_t = \{p_1, p_2, \ldots, p_N\} \tag{7}$$

where A_t denotes the action space at time step t and p_n identifies the lead vehicle priority on l_n (lane n) for N total number of approaching lanes which is a fixed-value for a given intersection i.e. N equals 8 for the intersection in Figure 12. The action $a_t = \max(A_t)$ is the selected action at timestep t which is essentially the vehicle with the highest p_n in A_t. The selected vehicle is then put into the priority assignment list for intersection crossing. There are two cases where certain actions can be masked out for selection; a) when there is no vehicle approaching the intersection on a particular lane, b) when there is no vehicle left to process on a particular lane for priority assignment. $A(s) \subseteq A$ denotes the set of masked actions in state s that are available for selection by the TCA.

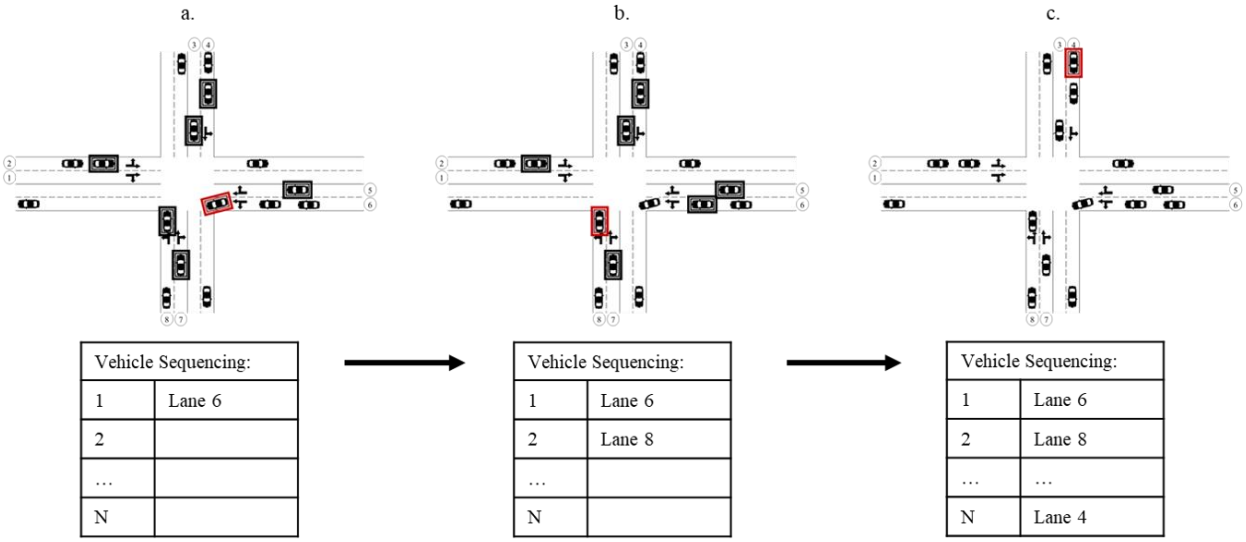


Figure 12 – Action selection is demonstrated through an example traffic flow at a 4-way junction. The vehicles with black frame are considered by the TCA for sequencing at that particular timestep, and the vehicles with red frame are the selected vehicles that have the next highest priority to cross the intersection after the TCA consideration.

The vehicle priority assignment via action selection is demonstrated with an example traffic flow at an intersection in Figure 12. The intersection has 8 approaching lanes, and in this case, $\mathrm{A} = \{ p_{\mathrm{i}} \}_{i=1}^{N}$. Let's assume that the vehicle priority list is empty at t_0 and all lanes apart from l_1 have approaching vehicles for intersection crossing where $\mathrm{A(s)} = \{ p_{\mathrm{i}} \}_{i=2}^{N}$. Therefore, the TCA will be restricted to select l_1. The vehicles highlighted with black frames in Figure 12.a are the vehicles to be considered next for the priority assignment and the TCA selection is l_6 which is highlighted with a red frame. The selected vehicle is put into the priority list accordingly. Bear in mind that one vehicle is selected at a time from all available lanes which is why all vehicles highlighted with black frames Figure 12.a are the same in Figure 12.b apart from l_6. The next vehicle in the queue on l_6 is now highlighted as the next vehicle in the queue. The priority assignment process is repeated until all vehicles are sequenced for intersection crossing. Figure 12.c shows the very last vehicle selected with TCA having $\mathrm{A(s)} = \{ p_4 \}$. In other words, all actions apart from l_4 are masked in the last step.

The vehicle priority assignment process is executed every T_CONTROL control interval (See Section 3.3.3) and the approaching vehicles with no priority yet are all buffered. It is important to note that the generated priority list of vehicles do not have any crossing time windows allocated yet. This will be done as part of conflict resolution stage that will be explained in Section 4.4. The role of AI in the unsignalised traffic control system is essentially to determine the priority of the vehicles for intersection crossing.

4.3.3. Reward Mechanism

Determining a reward mechanism is another challenge in the RL framework which has great impact on what an agent learns. Reward mechanisms should be structured in a way to encourage or discourage an agent on a selected action based on the objective of that agent. In other words, an objective function is encoded in the form of a reward function in the RL setting. A reward is a scalar value that represents how good or bad an action taken by an agent on a particular environment state and it depends on the selected action, current and next states of the environment in which an agent operates. This dependency can be shown as:

$$r_t = \mathrm{R}(s_t,\ a_t,\ s_{t+1}) \tag{8}$$

where, at timestep *t*, r_t is the reward, a_t is the action taken, s_t is the current state and s_{t+1} is the next state. The goal of an agent is to get as high a cumulative reward as possible over a horizon as it correlates to going in the right direction of achieving the objective of that agent. The aforementioned horizon can be finite (i.e. computer or board games etc.) or infinite as in this research work in which the traffic

control system is in operation continuously. Cumulative reward over an infinite horizon is intractable. Hence, a discount factor is applied to the cumulative reward to make it mathematically convenient and it refers to how significant the future rewards are with respect to the current state. The discounted cumulative reward which is also called discounted return over an infinite horizon is:

$$\mathrm{R}(\tau) = \sum_{t=0}^{\infty} \gamma^t r_t \tag{9}$$

where $\gamma \in (0,1)$ is the discount factor and τ is a sequence of states and actions. Many different objectives can be considered when defining the reward for a traffic control application, and these objectives can include journey time, junction queue waiting time, junction throughput, preventing stop-and-go movements, accident avoidance and fuel consumption. In this research work, the objective is to reduce traffic congestion and it is related to reducing the vehicle delay times during intersection approach and crossing. The reward for the TCA at timestep *t* is a weighted sum of three factors. These reward terms are decided based on the initial experiments in the simulation tool with various different weight factors and reward terms.

$$r_1 = \frac{\frac{T_{max}}{2} - t_{n\_delay\_max}}{T_{max}} \tag{10}$$

$$r_2 = \frac{\frac{T_{max}}{2} - t_{n\_masked\_delay\_max}}{T_{max}} \tag{11}$$

$$r_3 = \begin{cases} 0, & if\ traj_{a_t-1} \cap traj_{a_t} = \emptyset \\ -\frac{1}{k_n}, & otherwise \end{cases} \tag{12}$$

from which, the final reward value is obtained as a weighted sum of the reward terms in Eq. 10-12.

$$r_t = w_1 * r_1 + w_2 * r_2 + w_3 * r_3 \tag{13}$$

where r_t is clipped within [-1, 1] range. In Eq. 10, r_1 term gives more reward as the average vehicle delay times get smaller on all lanes where $t_{n\_delay\_max}$ represents the maximum of average vehicle delays on all lanes (same as t_{delay} in Section 4.3.1 in state representation) and T_{max} is a fixed configuration value to normalise the reward term. r_1 essentially ensures that all lanes have equal importance in reducing overall congestion at the intersection. In Eq. 11, r_2 term gives more reward

when the most congested lane is prioritised for vehicle sequencing. $t_{n\_masked\_delay\_max}$ denotes the maximum of the average vehicle delays excluding the masked lanes. In other words, if TCA assigns vehicle priorities starting from the most congested lane to the least, r_2 term will increase. r_3 in Eq. 12 is actually a negative reward (also known as a punishment term) to discourage frequent lane switch in the priority assignment. r_3 will decay exponentially when more vehicles are selected in a row from the same approaching lane, meaning less punishment for TCA. For r_3, an exponential decay term is used instead of a linear decay as it resulted in less congestion and vehicle delays during the initial experiments in the simulation tool.

Traffic control decisions can cause deviation from equilibrium in traffic flow (for instance, stopping the lead vehicle of a platoon at CrA entry point), and this perturbation propagates along the stretch of an approaching lane gradually. This phenomenon is called shockwave formation (FHWA, 2001). With this in mind, it becomes more important for the reward mechanism to capture the average traffic flow behaviour and change in time as a direct mid- to long-term consequence of the action selections made by the TCA.

4.3.4. Neural Network

The neural network for the proposed traffic control application consists of multiple layers. The input layer is the first layer which receives the observation vector, and the data is passed forward to the hidden layers where feature extraction happens. The hidden layers are a mixture of Fully Connected (FC) and Long Short Term Memory (LSTM) layers that extract some spatio-temporal features about the traffic environment. The final layer is the output layer that produces the vehicle priorities for each approaching lane.

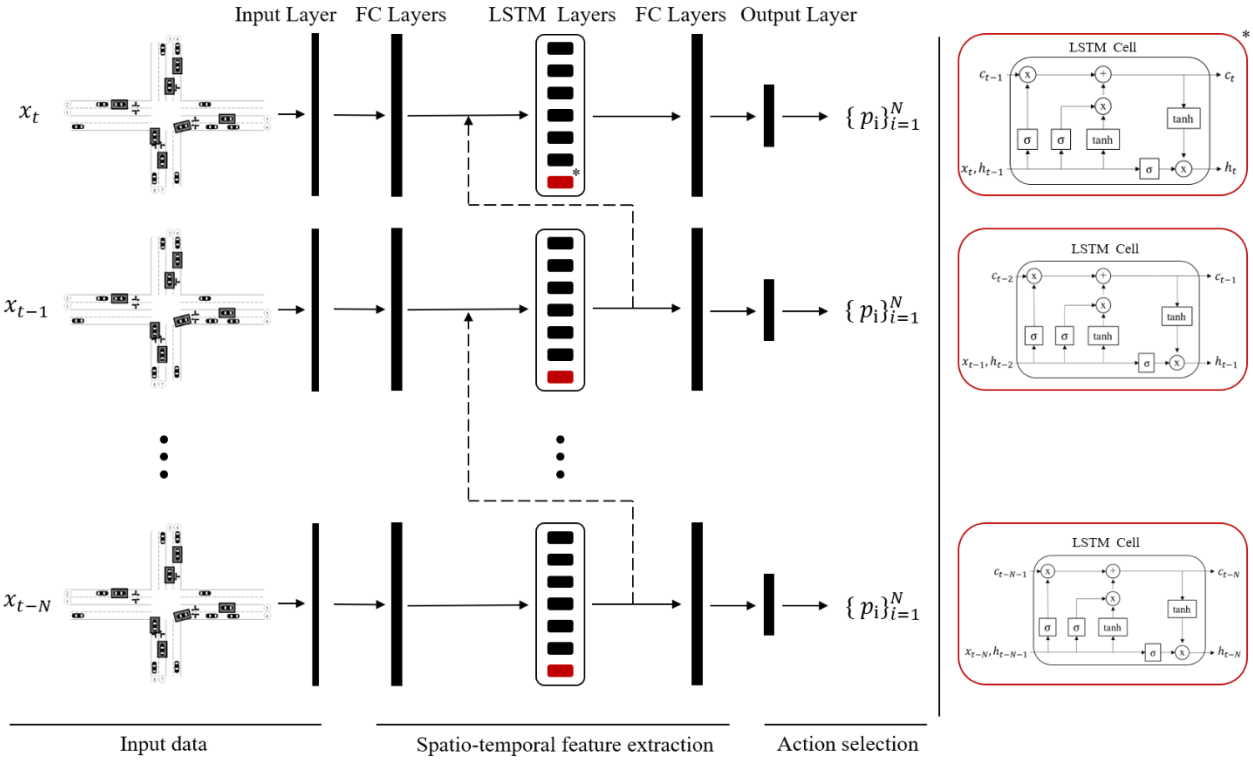


Figure 13 – Neural network setup with LSTM layers

Figure 13 shows the details of the neural network in which x_t is the observation vector at timestep *t*. The objective of FC layers in the network is to extract spatial features i.e. the hidden relationship between vehicle positions, speeds, distances etc. On the other hand, LSTM layers extract the temporal features i.e. the effect of previous outputs on the current state. Hence, there exists the LSTM connection with dashed lines in Figure 13 between timesteps.

A closer look at the structure of the LSTM cells for each time step is shown in red boxes in Figure 13. At timestep *t*, the LSTM unit gets the previous cell state vector c_{t-1} and hidden state vector h_{t-1} as an input together with the observation vector x_t. The information flow inside the LSTM cell is regulated via three gates, namely input, output and forget gates. These gates are composed of sigmoid functions and pointwise product blocks in order to control to which extent the current input data should be remembered or forgotten in the next timestep.

4.3.5. Policy

In RL, the policy can be defined as the strategy that an agent adopts in order to achieve its goals. With this in mind, the policy brings together the state representation, action space, reward mechanism and the neural network under Markov Decision Process (MDP) framework (Sutton and Barto, 2018). The

policy determines the way an agent behaves at a given time in the environment by having a probability distribution over the action space for the environment states.

A policy π can formally be structured as a tuple of the form *(S, A, P, R)* where *S* is the state representation, *A* is the action space, *P* is the probability matrix of transition from one state to another, and finally, *R* is the reward mechanism. A policy in RL is parameterized via the neuron weights and biases of the neural network, and this is done via an optimisation process during the training session which will be explained in detail in Chapter 5.

In this research work, the policy is used in the context of the actor-critic architecture (Sutton and Barto, 2018) in which the actor essentially updates the policy parameter set, θ, for $\pi_\theta(a|s)$ as guided by the critic. The value of an action-state pair, $Q^\pi(s,a)$, when started with a random action in state *s* and acted according to policy π afterwards is defined as below and it is also called as the expected return:

$$Q^\pi(s,a) = \mathrm{E}\,[\mathrm{R}(\tau)|s_0 = \mathrm{s},\ a_0 = \mathrm{a}] \tag{14}$$

where R(τ) is the sum of discounted rewards from Eq. 9. $Q^\pi(s,a)$ can also be interpreted as the expected cumulative future reward. When the optimal policy (the best strategy that leads the agent to achieve its goals) is used by an agent, the expected return $Q^\pi(s,a)$ is maximised as optimal action decisions are taken every timestep. The optimal action-state pair value is given by:

$$Q^*(s,a) = \max_\pi\ Q^\pi(s,a) \tag{15}$$

It is also important to note that there might be multiple action sequences that lead to optimal value $Q^*(s,a)$. In that case, all of those action decision sequences are considered optimal. If $Q^*(s,a)$ is obtained, then in a given state, the optimal action to take is also found by solving the equation (Watkins, 1989) below:

$$a^*(s) = arg\max_a\ Q^*(s,a) \tag{16}$$

In Section 4.3.2, it is explained that the proposed traffic control action space is continuous. With this in mind, finding $a^*(s,a)$ among infinite action choices is not trivial and intractable as it requires

computing the Q-values for each possible action every timestep to determine which one is the optimal action. Therefore, in this research work, TD3 (Fujimoto, van Hoof and Meger, 2018) algorithm is utilised which is a modified version of DDPG (Lillicrap *et al.*, 2015) algorithm. TD3 overcomes the aforementioned challenge with continuous action spaces by using a gradient-based learning rule for a policy $\pi(s)$ that presumes $Q^*(s,a)$ is differentiable with respect to the action and the following approximation can be made (Sutton and Barto, 2018):

$$\max_a Q^*(s,a) \approx Q^*(s,\pi(s)) \tag{17}$$

TD3 algorithm is specifically developed for continuous action spaces as in this project and it is an off-policy algorithm, meaning the TCA have the ability to learn from historical data obtained from the traffic environment. Chapter 5 will give more details on how this algorithm is used to train an agent for the unsignalised traffic control application.

4.4. Conflict Resolution of the Shared Intersection Space and Time for Vehicle Crossing

The problem statement given in Chapter 2 considers a scenario where multiple vehicles approach a traffic intersection from multiple lanes. Safe intersection crossing in the absence of a traffic light requires a vehicle trajectory conflict management method to avoid collisions. This section will give details on the developed conflict management method which takes the vehicle priority list generated by the AI agent as an input and produces crossing time windows for all vehicles in the queue as an output. The proposed vehicle trajectory conflict management method is derived from a study by Levin and Rey (2017) in terms of the conflict point modelling and travel time estimations of vehicles through those CPs.

4.4.1. Vehicle Trajectory Conflicts

In the trajectory conflict stage, the primary focus is the prevention of the side collisions at CPs. Transportation institutions worldwide release guidelines and manuals on various aspects of road networks in order or standardise and synchronise activities across the country. With this in mind, CPs are also determined by these institutions for different intersection geometries, and FHWA (2004) is an example guidelines document by Federal Highway Administration in the USA that describes the CPs for a given intersection.

An example 4-way intersection (left-hand side drive) is shown in Figure 14 where there are three types of CPs, crossing, merge and diverge as shown with different colours inside the CrA. There are 32 CPs in total where there is a risk of lateral vehicle collision. The objective of the conflict management method is to schedule vehicles crossing with time windows so that only one vehicle occupies each CP at any given time.

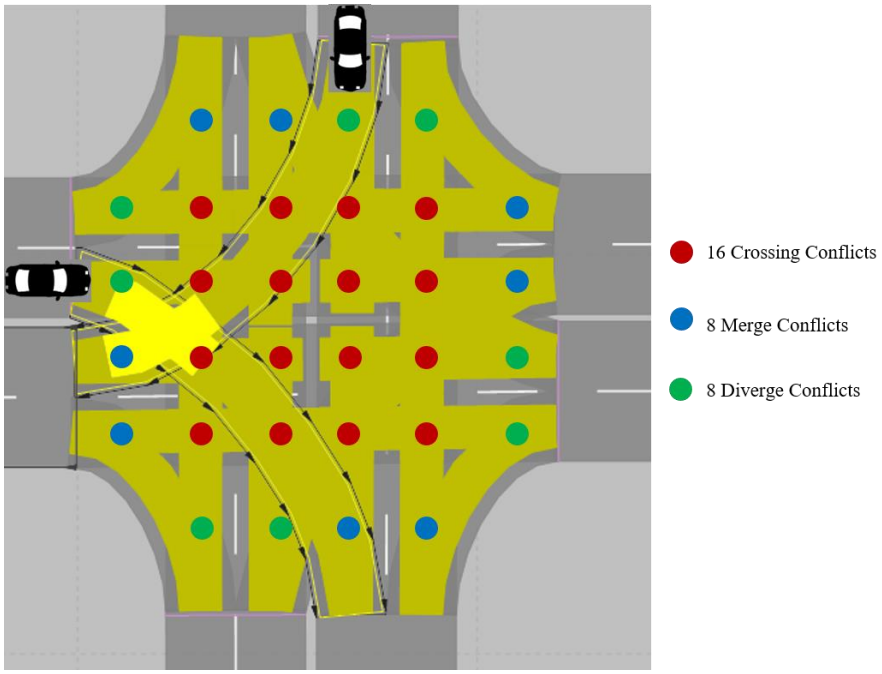


Figure 14 – Vehicle trajectory CPs inside the CrA

Each approaching vehicle i has a pre-determined trajectory $traj_i$ through the intersection which is represented as an ordered list of CPs. For each crossing vehicle i, $t_i(p)$ must be determined which is the time that vehicle i occupies CP p where $p \in traj_i$. For the sake of simplicity, the vehicle type (i.e. van, truck etc.) is assumed to be a passenger car and the vehicle speed during intersection crossing, v_{cross}, is assumed to be uniform which is decided by the vehicle and communicated to TCA via V2I communication as explained in Chapter 2. If a vehicle is entering the CrA from an initially stopped condition at the entry point of the intersection, then the vehicle acceleration, a_{cross}, is assumed to be uniform until v_{cross} is reached. Even though a_{cross} and v_{cross} are assumed to be uniform, a safety buffer of $\pm\Delta v_{cross}$ and $\pm\Delta a_{cross}$ is added to the calculations which essentially ensures that there is room for error as vehicles cross the intersection and there is sufficient safe spatio-temporal space between vehicles with conflicting trajectories.

4.4.2. Trajectory Conflict Table

There is a fixed number of CPs for a given intersection geometry. With this in mind, this research work uses a Trajectory Conflict Table (TCT) to track the CP occupation for each crossing vehicle per timestep. Let us imagine a scenario where there is only one vehicle requesting to cross an intersection similar to Figure 15. In this case, the vehicle $traj_i$ is composed of 6 CPs. Figure 15.a shows the vehicle crossing on the road whilst Figure 15.b highlighting with red dots the conflict table state as the vehicle moves from one CP to another.

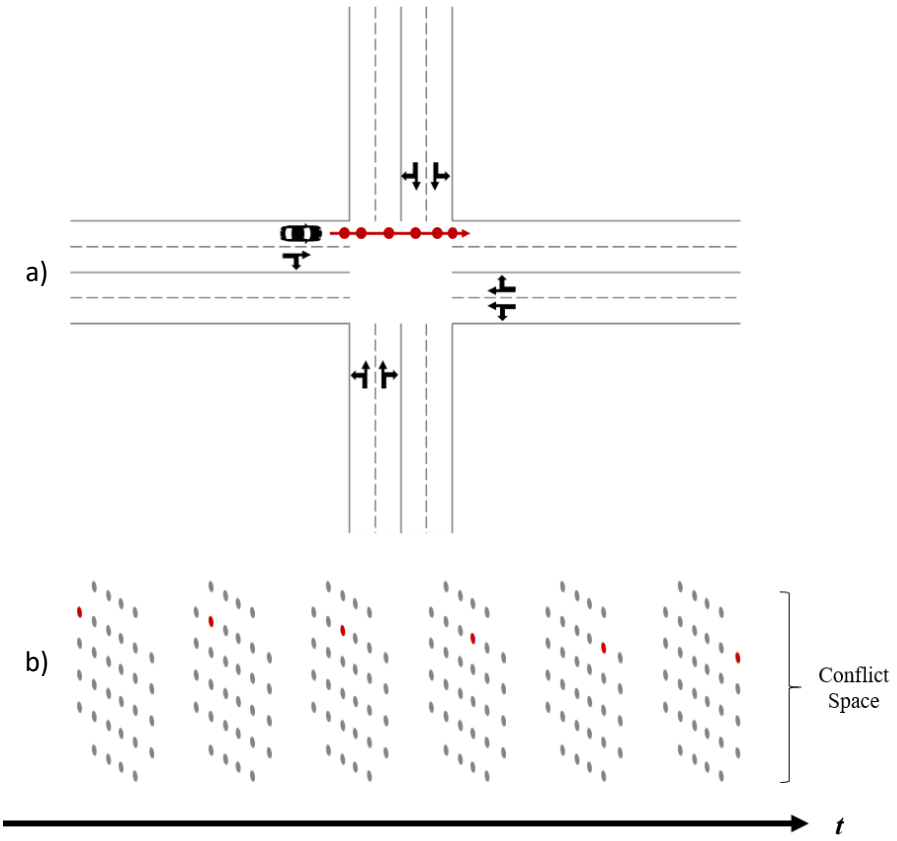


Figure 15 – a) Vehicle trajectory through the intersection is shown with a set of conflict points that must be occupied by a single vehicle at any moment in time, b) Conflict points occupation by the crossing vehicle is shown over time.

The TCT approach for collision avoidance and vehicle conflict resolution brings some advantages. 1) The approaching vehicles may change their trajectory decisions after securing a crossing time window (i.e. lane change, exit lane change, crossing speed update etc.). When that happens, TCT can facilitate allocation of an alternative crossing time window and updating the CP occupation times. 2) The fact that the TCT has a fixed number of CPs offers a deterministic processing time when deployed in the field.

4.4.3. Vehicle Crossing Time Allocation

Vehicle crossing time start and duration times are calculated based on the vehicle trajectory $traj_i$ through the intersection. Firstly, the travel time between CPs for a given vehicle is calculated. For vehicle *i*, the travel time t_{trav_i} constraints between two consecutive CPs $p_1, p_2 \in traj_i$ with a distance of $d_{cp_i}(p_1, p_2)$ considering $\pm\Delta v_{cross}$ safety buffer is given by:

$$\frac{d_{cp_i}(p_1,p_2)}{v_{cross} - \Delta v_{cross}} \leq t_{trav_i} \leq \frac{d_{cp_i}(p_1,p_2)}{v_{cross} + \Delta v_{cross}} \tag{18}$$

Once t_{trav_i} is calculated between all $traj_i = \{p\}_{j=0}^{K}$ where K is the total number of CPs along the trajectory of vehicle *i*, a suitable crossing time window search can be done in the TCT.

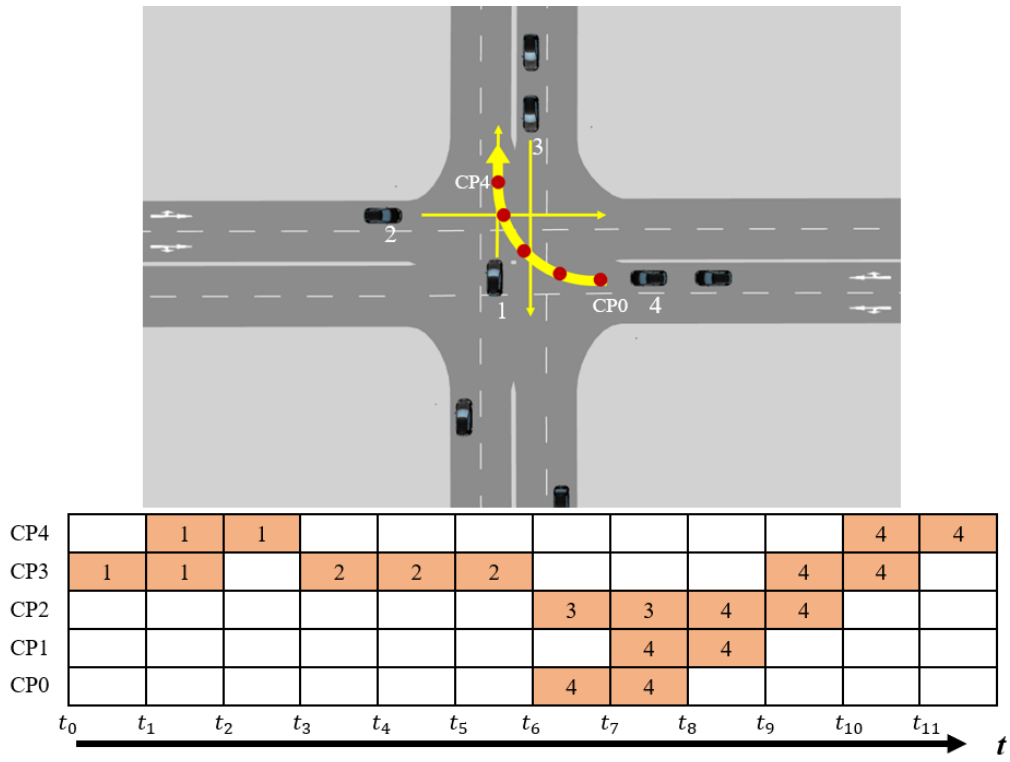


Figure 16 – Vehicle trajectory conflict resolution table for crossing time allocation.

The crossing time window allocation will be shown with an example intersection crossing scenario as in Figure 16. Let us assume that Vehicle 1, 2 and 3 have already been allocated crossing time windows and Vehicle 4 is the next vehicle to consider. Vehicle 4 has $traj_i = \{p\}_{j=0}^{K=5}$ and CPs are shown with red dots starting from CP0 to CP4 on its trajectory through the intersection. Even though, Vehicle 4 is ready to cross at t_0, it is not allowed to cross as there are trajectory conflicts with Vehicle 1 on CP3-4,

with Vehicle 2 on CP3 and with Vehicle 3 on CP2. Therefore, the first suitable crossing window for Vehicle 4 starts from t_6 which is the crossing start time that the vehicle is allowed to enter the CrA for crossing.

4.5. Summary

In this chapter, the details of the AI algorithm and methods are presented in the context of an unsignalised traffic control. Traffic is stochastic in nature, meaning the reaction of the environment might not be predicted precisely. In addition, traffic environment is one of the prime examples where the traffic control actions affect the flow of vehicles gradually in time rather than immediately after taking a particular action. This led this research work to explore and implement methods that can handle stochasticity and delayed outcome.

The state representation, the action space and the reward mechanism are explained, all of which will be combined together under TD3 algorithm in the next chapter. This algorithm is off-policy and has the ability to handle continuous action spaces. The parameterisation of the AI model will be done in the next chapter during what is called a training session. The main task of the AI algorithm is to determine the vehicle priorities every control cycle.

The vehicle trajectory conflict management and the crossing time window generation processes are separated from the AI algorithm. The rationale is to differentiate achieving the objectives of traffic congestion reduction and ensuring safety via two independent processes.

Chapter 5

5. Algorithm Training: AI Traffic Control in Computer Simulations

5.1. Introduction

The term training, in the context of AI, is used to define the learning procedure by using an optimisation algorithm. There are many different optimization algorithms, among which the most popular ones are gradient descent (Curry, 1944), conjugate gradient (Hestenes and Stiefel, 1952), quasi-newton method (Wright and Nocedal, 1999), levenberg-marquardt (Levenberg, 1944), stochastic gradient descent (Robbins and Monro, 1951) and adaptive linear momentum (Kingma and Ba, 2014). The objective of the training is to achieve the minimum loss possible which can be translated into obtaining the maximum possible cumulative reward for the RL algorithm. This is essentially done in an iterative way during training by searching for an optimum parameter set that fits the neural network to the input data from the environment.

Training an agent to achieve the objective of controlling traffic at intersections with minimum vehicle delays requires a simulation environment so that the agent can learn by trial-and-error as it interacts with the environment. In this research work, Vissim simulation tool from PTV Group (PTV – Planung Transport Verkehr AG, 2019) was used. The rest of this chapter explains the details of the training process and how the Vissim traffic model was setup as the control environment.

5.2. Training Methodology

The learning problem for the traffic control task is formulated in terms of a loss factor minimisation by using the adaptive linear momentum (Kingma and Ba, 2014) optimisation algorithm, hereafter referred to as the Adam optimizer. The Adam optimizer is widely used in machine learning applications and it has been chosen mainly due to its computational efficiency, suitability for problems with noisy and sparse gradients and small memory space requirements when coded in software.

The Adam optimizer executes a search through the neural network parameter space in order to decrease the loss at every epoch, which refers to one full cycle through the training data, and it is done by adjusting the neural network parameters. In particular, Adam optimizer calculates the exponential moving average of the gradient and the squared gradient when determining the parameter adjustment rate and direction. At first, a neural network is initialised with a random parameter set and it is updated every epoch in the direction that minimizes the loss value until a training stop criterion is reached i.e. the loss decrement in one epoch reaches a plateau.

5.2.1. Training Steps

In this section, the training procedure is explained step-by-step and the TD3 algorithm breakdown is presented. Figure 17 shows an overview of the forward pass and back propagation stages. Forward pass refers to the process of obtaining the output layer data after the traffic observation vector is given as an input to the neural network during which the data cascades through the network layers. Back propagation, on the other hand, refers to the process of neural network parameter update via the Adam optimizer.

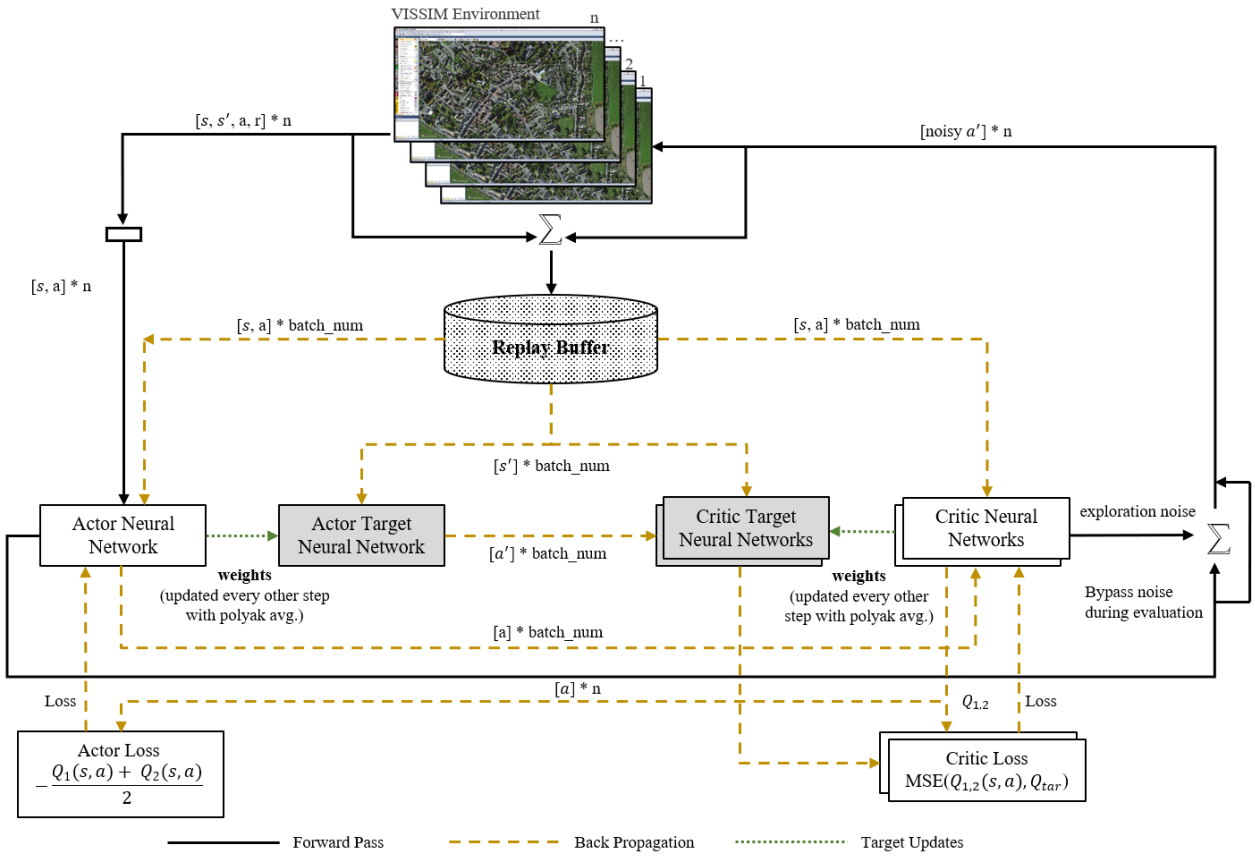


Figure 17 – TD3 algorithm overview during forward pass and back propagation

The training steps can be given as below:

Step 1:

The replay buffer is initialised as the first step. As mentioned in Chapter 4, the TD3 algorithm is off-policy, and it uses a replay buffer to store past transitions when executing the policy in Vissim traffic environment. Each transition can be represented as a tuple in the form of (s, s', a, r) where, s and s' are the current and next observations, a is the selected action and r is the reward value. During training, the transition tuples in the replay buffer are queried to essentially replay the agent's experience either in a shuffled way or in the original order they are stored.

Step 2:

The neural networks (See Section 4.3.4) are built as a next step by using TensorFlow (Abadi *et al.*, 2016) open source machine learning libraries. There are 6 neural networks in total each with exactly the same number of layers and neurons, one each for the Actor and Actor Target models and two each for the Critic and Critic Target models as shown in Figure 17. The actor networks learn the policy $\pi(s|a)$ while the critics learn $Q^{\pi}(s, a)$.

The rationale behind having target networks for both actor and critic is to be more conservative when updating the neural network parameters. In other words, the target network parameters are constrained to change at a slower rate which is determined by polyak averaging (Fujimoto, van Hoof and Meger, 2018):

$$\emptyset_{target} = \rho\emptyset_{target} + (1 - \rho)\emptyset \tag{19}$$

Where ρ is in the range [0,1] and determines the rate of change in the target network parameter set.

Step 3:

The agent starts taking actions according to the initial policy in the Vissim traffic environment. Forward pass is repeated until the replay buffer is full after which *batch_size* number of transitions are sampled from the buffer. For each transition in the sampled batch, the Actor target produces the next action a' as shown in Figure 17, and (s', a') pair is given as an input to two Critic targets. The Critic targets return the values of the state-action pair, $Q_{tar1}(s', a')$ and $Q_{tar2}(s', a')$, independent from each other. The final Critic target Q-value is obtained by:

$$Q_{tar} = \mathrm{r} + \gamma(\min(Q_{tar1}, Q_{tar2})) \quad (20)$$

where γ is the discount factor in the range [0, 1]. Taking the minimum of two Q-values has been found to stabilize the optimisation process (Fujimoto, van Hoof and Meger, 2018) as optimistic Q-value estimates are avoided by ignoring the higher Q_{tar} value. Following this, the two Critic networks take (s, a) pair as an input and produces $Q_1(s, a)$ and $Q_2(s, a)$ in order to compute the final Critic loss:

$$Loss_{critic} = \mathrm{MSE}(Q_1(s, a), Q_{tar}) + \mathrm{MSE}(Q_2(s, a), Q_{tar}) \quad (21)$$

where MSE refers to mean-squared error loss. This loss value is used during backpropagation to update the Critic network parameters. This training step where the agent tries to reduce the Critic loss is called the Q-learning. Q-learning step aims to find the optimal parameter set for the Critic networks. The next step moves on to policy learning step.

Step 4:

Policy learning aims to find the optimal parameter set for the Actor network in order to maximise the expected return (See Eq. 14). Based on the approximation made in Eq. 17, the Q-value from the Critics is correlated with the expected return, meaning as the Q-value is increased, the expected return goes towards being optimal. In this case, the loss for the policy learning is the mean value of the Q-values from the Critics:

$$Loss_{actor} = -\frac{Q_1(s, a) + Q_2(s, a)}{2} \quad (22)$$

During backpropagation, gradient ascent is used, hence the negative sign in Eq. 22, by differentiating the actor loss with respect to the Actor network parameters in the direction that maximises the expected return.

The important point to note here is that policy learning is done every other step whereas Q-learning in Step 3 is done every step. If the Q-learning is poor, the policy becomes poor as well, and it can cause divergence of the loss moving towards minima. It is why Q-learning is done at double the rate of policy learning to increase the performance of convergence to the optimal parameter set.

Step 5:

The final step in the training cycle is to do an update on the target network parameters of the Actor and the Critic which has not been done up until this point. The target network parameters are updated based on Eq. 19 which is also called a soft update, and it essentially copies the weights of the Actor and Critic networks with polyak averaging into the target networks. Similar to the policy learning, the target network updates are done every other step to improve training performance stability.

Steps 2-5 are repeated until any of the training stop conditions are met as previously explained. The replay buffer is overwritten with the new transitions starting from the oldest entry as the agent continues to operate in the Vissim environment.

5.2.2. Exploration-Exploitation

The exploration-exploitation dilemma in RL refers to the trade-off that an agent makes when taking actions in the environment in terms of exploring the new actions in the action space versus exploiting the previously gained knowledge about certain actions. Therefore, it is crucial in RL problems to set the ratio of exploration to exploitation appropriately.

There were three techniques applied in this research work to increase exploration whilst ensuring that the agent also exploited the good actions when required:

- The authors of the TD3 algorithm (Fujimoto, Van Hoof and Meger, 2018) adds a mean-zero Gaussian noise to the target actions a' as in training step 3 in the previous section prior to provide the tuple of (s', a') to Critic target networks. This noise is found to be useful in exploring new actions. In this research work, the scale of noise over the course of training was reduced linearly rather than keeping it fixed as in the original paper. This meant that the agent explored the new actions less as training progressed.

- A mean-zero Gaussian noise, exploration noise, was also added to the actions during forward pass and the scale of the noise was reduced linearly over the course of training.

- At the beginning of the training, the replay buffer was originally filled with data by the agent taking actions according to the initial policy. This was essentially equivalent to taking random actions from which most of the action sequences did not yield high rewards. Therefore, in this

work, the fixed-time traffic light policy was utilised when taking actions until the replay buffer was full after which the agent followed its own policy. This was found to be useful in terms of reducing the training time.

5.2.3. Training Parameters

The training procedure was executed with 4 environments that were independent from each other as shown in Figure 17. The reason for using 4 environments was due to the Vissim licence restrictions where more than 4 instances of the software tool were not possible to run in parallel. It meant that the replay buffer contained experiences of multiple agents. This is a common technique in RL training to speed up the learning process and to reduce correlation between transitions. During backpropagation, the global gradient was calculated by averaging all local gradients of individual agents.

The hyperparameters for the agent and the training procedure are given in Table 3. These hyperparameter values was decided after executing a manual tuning process in which each hyperparameter was changed one at a time to observe their effect on the training process.

Hyperparameter	**Value**
FC layer 1 size	128
FC layer 2 size	128
LSTM layer size	64
FC layer 3 size	64
FC layer 4 size	32
FC layers activation	ReLu
LSTM layer activation	Tanh
Discount factor γ	0.998
Polyak averaging ρ	0.05
Learning rate actor	$1e^{-4}$
Learning rate critic	$1e^{-3}$
Target action a' noise initial scale	0.15
Target action a' noise decay steps	$15 * 10^3$
Exploration noise initial scale	0.2
Exploration noise decay steps	$15 * 10^3$
Batch size	256
Replay buffer size	$50 * 10^3$
Adam optimizer	$\epsilon = 10^{-8}, \beta_1$

Table 3 – The hyperparameter list

Reward mechanism parameter T_{max} was set to 60 seconds and the weights for the reward terms (See Eq. 13) were $w_1 = 1.0$, $w_2 = 1.0$ and $w_3 = 0.5$.

5.3. Traffic Environment in Simulation

Many RL applications, that are deployed in the field, require a simulation platform for the training procedure as it is not safe for an agent to learn by trial-and-error in a real-world setting. The Vissim traffic simulation tool was adopted in this research work to proxy the effects of unsignalised traffic control on traffic flow and congestion under mixed-driving conditions where CHVs and CAVs at SAE Level 5 co-existed in traffic.

The core principles of how CAVs may impact traffic flow at intersection crossings are related to the configuration of these vehicles including but not limited to wireless connectivity, longitudinal and lateral motion control behaviour and gap acceptance times. Therefore, it is essential to let the agent operate under as many different traffic conditions and driving behaviours as possible during the training procedure so that the trained agent parameters can be scaled and deployed in a real-world setting.

5.3.1. Introduction to Vissim

Vissim simulation tool is a microscopic, time-step oriented, and behaviour based simulation tool for modelling traffic in urban and rural settings, pedestrians, public transportation as well as rail transportation. The interaction between each element can be modelled and simulated. The simulation resolution is configurable in Vissim which determines how many times, in a simulation second, data can be exchanged between the vehicles and the traffic control algorithm. In this research work, the tool was run at 10 Hz speed throughout all experiments which corresponds to 10 data exchange in a simulation second.

In Vissim, lateral and/or longitudinal control of multiple vehicles is possible. Parameter sets can be identified that allow for the representation of different driving behaviours in the traffic flow. With regards to the wireless communication, there is no built-in model in Vissim for V2X communications. However, it is possible to integrate external tools via Component Object Model (COM) interface for additional functionality such as V2X communications.

The aforementioned COM interface also allows write or read access to all simulation data and parameters from the external software while simulation is running. The data available from this interface includes but not limited to the road network parameters, location and speed of all vehicles. This ability of Vissim is a particular interest in this research work in order to control the behaviour of the vehicles dynamically at every control cycle dependent on the unsignalised traffic control strategy. For example, a vehicle approaching an intersection for crossing may be requested to wait or slow down if the TCA gives priority to the vehicles on other approaching lanes. Further details on the implementation of the AI traffic control algorithm and its integration with the Vissim tool will be given in Section 5.3.5.

5.3.2. Road Network Setup

Road networks in the real-world contain several stochastic elements, and in many cases, there are multiple traffic control systems on the stretch of a link where roads with different capacities intersect and have an impact on each other. In this work, a 4-way intersection with two lanes on each approach and exit link was modelled as shown in Figure 18 in order to focus on the impacts of the proposed control method on traffic flow in an isolated way from other factors.

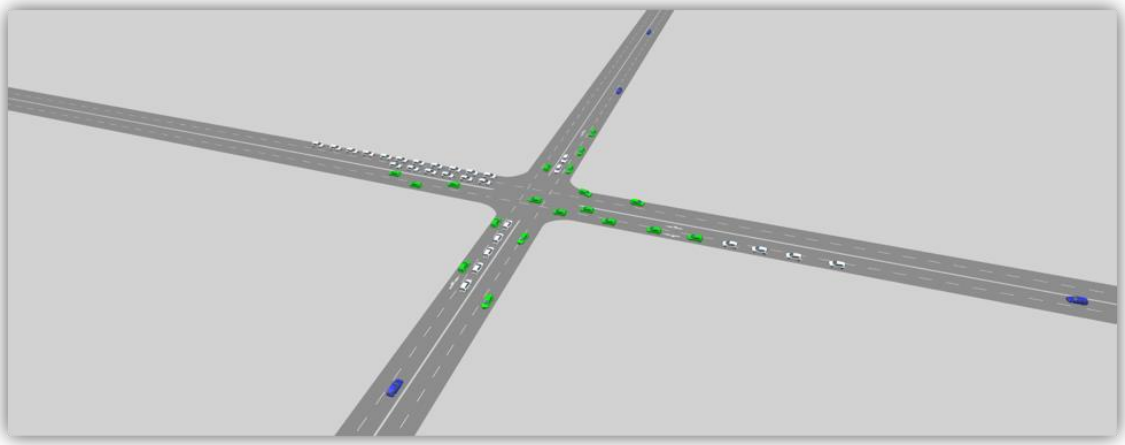


Figure 18 – The road network setup in Vissim traffic simulation for the training procedure

The modelled approach and exit links had a length of 600m and a speed limit of 50 km/h. The RSU with V2I capabilities was located near the shared space of the intersection area and it had a communication radius of 150m. The vehicles did not interact with any other infrastructure other than the TCA which operated inside the RSU.

The simulated road network was constructed as per the UK traffic rules and standards i.e. left-hand drive etc. However, the simulation results and data obtained are still relevant for other countries mainly due to the fact that an isolated intersection was studied with similar traffic flow and driving behaviour worldwide.

5.3.3. Driving Behaviour Generation

The driving behaviour in Vissim is based on the Wiedemann-99 car following model (Wiedemann and Reiter, 1992) and the parameters of this model can be adjusted to create different driving behaviours. The approach taken in this work was to have one general CHV behaviour and four different CAV driving behaviours (CAV B1, B2, B3 and B4) by systematically varying the parameters to enable CAVs to be more cautious or assertive than CHVs. To this end, the aforementioned parameter sets are based on the research work by Atkins (2016) in which the impacts of CAVs on traffic flow are analysed in detail in Vissim. There are 9 parameters in total as listed in Table 4 for each driving behaviour. It should be noted here that all CAV driving behaviours are still considered SAE Level 5 with different driving styles and CHV is considered SAE Level 0.

Param.	Description	CHV	CAV B1	CAV B2	CAV B3	CAV B4
CC0	Desired standstill distance between vehicles (m)	1.5	1.0	0.9	0.6	0.5
CC1	Headway time from the vehicle in front (s)	0.9	0.8	0.7	0.6	0.5
CC2	Headway longitudinal distance oscillation (m)	4.0	0.0	0.0	0.0	0.0
CC3	Time to recognise a preceding slower vehicle (s)	8.0	8.0	8.0	8.0	8.0
CC4	Negative desired speed difference (m/s)	0.35	0.05	0.05	0.05	0.05
CC5	Positive desired speed difference (m/s)	0.35	0.05	0.05	0.05	0.05
CC6	Influence of vehicle distance on speed oscillation	0.0	0.0	0.0	0.0	0.0
CC7	Oscillation during acceleration (m/s2)	0.25	0.3	0.35	0.40	0.45
CC8	Acceleration when starting from standstill (m/s2)	3.5	3.6	3.7	3.8	3.9
CC9	Acceleration at 80 kph (m/s2)	1.5	1.6	1.7	1.8	1.9

Table 4 – Driving behaviour parameters for connected human-driven vehicles and automated vehicles.

The rationale behind simulating multiple CAV driving behaviours is based on the assumption that, in the future, CAV manufacturers will enable end-users to set the driving behaviour as an option to suit their needs. In Table 4, the CAV driving behaviour becomes more aggressive from B1 to B4, meaning higher acceleration rate, shorter gaps between vehicles etc. CAV B1 is parameterised in a way to make

it more cautious then CHV whereas CAV B2-B4 are more assertive than CHV. Atkins (2016b) argues that the focus for mixed-fleet simulation models should not be on the fidelity of the CHV driving behaviour as it has been studied and understood in the literature, but the changes CAVs imply once deployed.

5.3.4. Traffic Demand and Vehicle Routes

Traffic demand and vehicle route choices are another configuration set in Vissim that is essential to vary during training for an agent to experience the potential impacts of CAVs under different operational scenarios i.e. peak time, off-peak time etc. In order to achieve that, three levels of traffic demand were parameterised to represent low (500 veh/h), medium (1000 veh/h) and high volume of traffic (2000 veh/h) similar to Atkins (2016). The traffic demand levels were changed during training at set intervals which was set as 2 hours.

Vehicle routes are represented as a fixed sequence of links and lanes in Vissim that a vehicle is requested to follow on the road network. The turning decision at intersection, whether to go left, right or straight, depends on this configuration. The relative ratio of turning decisions were set as 0.33 for left turn, 0.33 for right turn and 0.66 for straight during training. The vehicles that would turn left and right position themselves on the inner or outer lane as they approached the intersection as per the road markings whereas the vehicles that went straight could be on either lane.

5.3.5. Software Tools Integration

The complex nature of the proposed traffic control system required multiple software tools to be integrated as no single tool solution existed, at the time of writing this thesis, for the AI traffic control algorithm implementation and simulations. The interface diagram between the tools and the software components are shown in Figure 19. As explained previously, there are multiple agents running in parallel during training and collecting experiences independent from each other.

To this end, Vissim traffic environment and TensorFlow based AI model run in parallel processes having an exact copy of parameters and computing resources. V2I connectivity and driving behaviour control was implemented in the C++ programming language with an interface to Vissim in the form of a Dynamic Link Library (DLL) file. During training, the DLL file was called every 100 ms for each vehicle in the network.

The National Instruments (NI) LabVIEW tool was also integrated to provide test automation, data analysis, visualisation and logging features. The NI LabVIEW tool can be considered as the central hub where the experiments were started and monitored during training. Please also note that the integration of this tool was optional, and it was done mainly due to the author's vast experience of using it in various other industrial projects. Otherwise, NI LabVIEW tool could be replaced by a Python-based custom code.

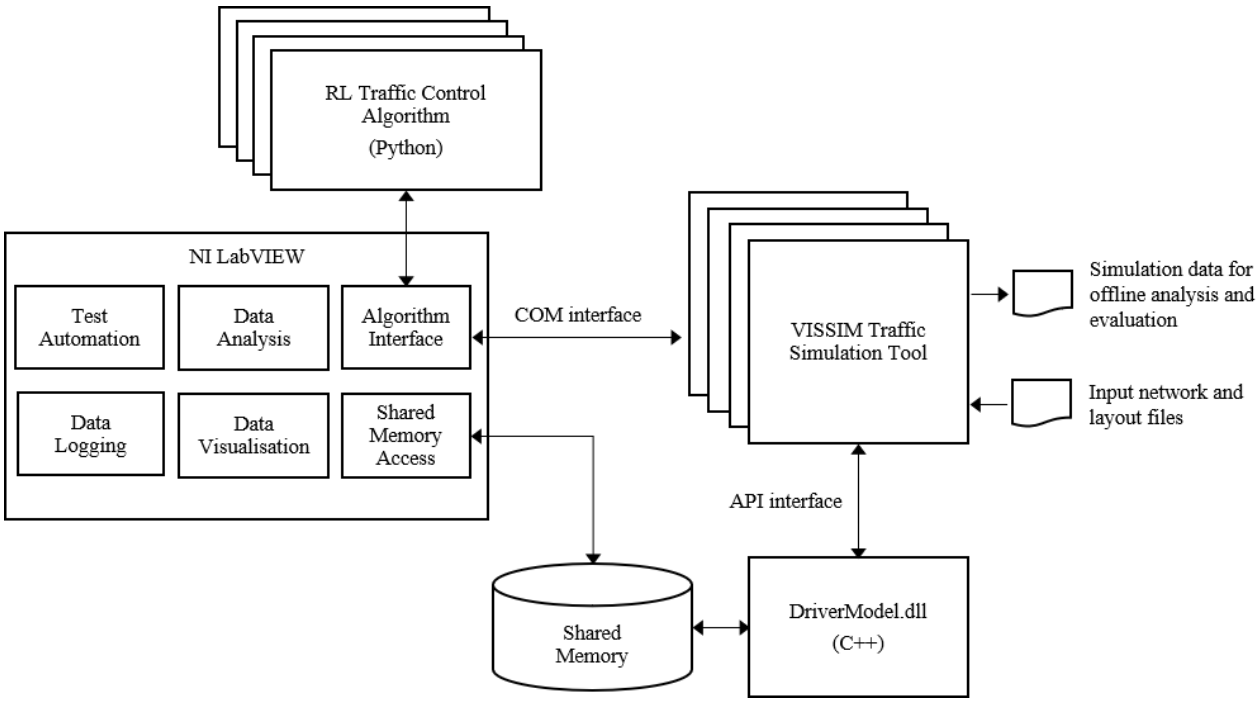


Figure 19 – The software tools setup integrating Vissim simulation tool, NI LabVIEW platform and the RL library.

5.4. Summary

In this section, the agent training methodology and the associated parameters and configurations have been explained in the context of the RL framework. The main objective of the training procedure was to enable the agent to experience all potentially possible situations in the traffic environment so that it could learn what action sequences result in better policy. Model-free RL algorithms like TD3 are sample-inefficient, meaning they require a lot of interactions to learn a good policy. The exploration and exploitation dilemma section has explained the techniques implemented in order to overcome the sample-inefficiency challenge and to reduce the training time. Filling the experience replay buffer initially with the traffic light policy has been found to be useful in terms of faster convergence to the optimal policy.

When the training is complete, the agent can be deployed at different traffic intersections as long as the number of approaching links to the intersection are the same as the training scenarios i.e. 4-way, 5-way etc. Otherwise, the agent has to be re-trained so that it can learn the optimum control policy with the target number of links on the road network.

The traffic simulation methodology and the tool chain have also been explained in this section. The penetration rate of CAVs is too small, worldwide as of today, to gather any real-world evidence about their impact on traffic flow. Therefore, any research work in this field of research utilises simulation tools to model their behaviour. In this work, this approach was also taken where various different driving styles were modelled in conjunction with traditional human driving behaviour in order to simulate mixed-driving scenarios. The selected tools represent the state-of-the-art solutions which are widely used within the machine learning and traffic engineering fields in academia and industry.

The next chapter focuses on the validation strategy of the proposed traffic control method which is also called the evaluation stage in RL. The trained agent operates under various different traffic conditions and the performance is compared against the traffic light based control method that exists in our road networks today. The software toolchain explained in this section is also used during the validation stage.

Chapter 6

6. Validation of AI Traffic Control in Computer Simulations

6.1. Introduction

RL algorithms can be evaluated based on how their policy performs in the environment after the training procedure is complete. The measure of performance is quantified with metrics that are specific to the traffic control task and the traffic environment. In general, when agents are trained on specific tasks, it is important to evaluate the capability of generalising the final policy to unseen situations in training. In this research, the TCA was validated under such traffic scenarios in Vissim to determine how well the agent generalised its traffic control policy. In this chapter, the details of the validation procedure will be given including the identified performance metrics and the generated traffic scenarios in Vissim.

6.2. Validation Scenarios

The previous chapters of this thesis have explained the objectives of the proposed unsignalised traffic control method and the mechanisms implemented that bring together wireless connectivity, operation of CAVs and the machine learning strategies. In the following sub-sections, these objectives will be translated into a methodological approach in which the impacts of the traffic control method can be measured and quantified.

To this end, multiple scenarios in the traffic environment are defined that involve the following key features:

- Road geometry i.e. roundabout, 4-way junction etc.
- Traffic demand range on the road network,
- Traffic demand ratio where intersecting roads have different demand levels,
- Driving behaviour from cautious to assertive,
- Mixed-driving where vehicles at different SAE Levels co-exist in traffic,
- Traffic control methods,

A combination of these key features was used in the validation scenarios to measure the impact of the traffic control methods on traffic flow and congestion. The main reason for validation via computer simulations in this work is to be able to easily change the physical properties and features of the testbed as listed above in order to ensure TCA performance is generalised to different traffic scenarios.

6.2.1. Road Geometry

Road geometry, in the context of intersection design, refers to the way that intersecting roads are connected to each other. There were two types of geometric designs considered for the validation procedure. The first one was a 4-way junction which was the exact replica of what was used during training, and the second one was a 4-way roundabout as shown in Figure 20. The allowed turning movements and the number of lanes on each link were the same for both scenarios. The conflict resolution for the roundabout has been updated from the 4-way junction as there are less CPs (8 in total) in the intersection CrA compared to 32 CPs in 4-way junction as shown in Figure 14.

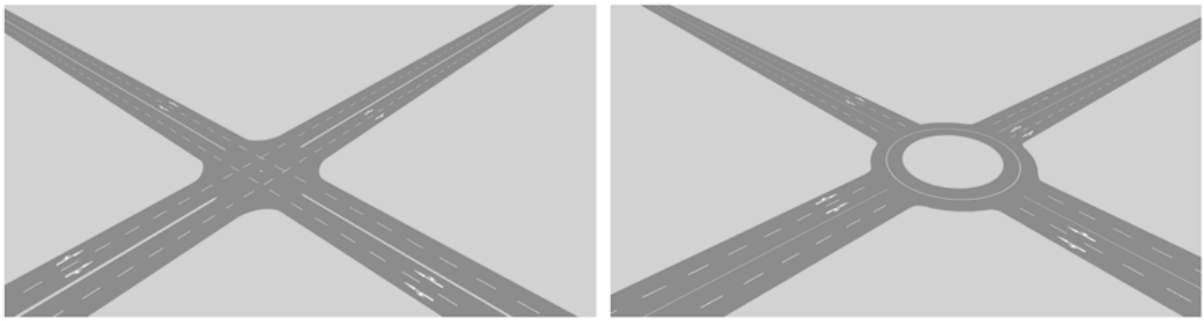


Figure 20 – Intersection road geometry types used during validation, 4-way junction (on the left), 4-way roundabout (on the right)

6.2.2. Traffic Demand

Traffic demand represents the number of vehicles per hour (veh/h) on the road network. Similar to the training scenarios (See Section 5.3.4), three levels of traffic demand were defined to represent low (500 veh/h), medium (1000 veh/h) and high (2000 veh/h) volumes of traffic. In Vissim, the time that a vehicle enters the road network is defined stochastically even though the traffic demand per hour is set as a fixed value. Firstly, an average time gap between two vehicles was calculated based on the defined traffic demand scenarios, which was then used as the average value of a negative exponential distribution. The actual time gaps between two vehicles entering the network were obtained from this distribution which relates to a Poisson distribution.

Additionally, two more traffic demand related scenarios were generated that represented situations where two links intersected with same or different traffic demand levels. The traffic demand ratio of 1.0 (Major link / Major link) and 2.0 (Major link / Minor link) were used which refer to the traffic demand ratios of the intersecting links. For example, if the North-South and the West-East links have traffic demands of 1000 veh/h, then the demand ratio is 1.0. Similarly, if one of the links has half the level of traffic demand i.e. 500 veh/h, then the demand ratio is 2.0.

6.2.3. Driving Behaviours

The driving behaviours generated for the training were used in the exact same way during evaluation. See Section 5.3.3 for a detailed explanation. In summary, there were 5 different driving behaviours, CHV and CAV B1 to B4.

6.2.4. Penetration Rate

Penetration rate is defined as the percentage of SAE Level 5 vehicles in the total vehicle fleet in the road network. Mixed-driving fleet operations will be the case in the near future until all vehicles are of type Level 5. To consider a variety of potential future cases, 5 levels of penetration rates were considered during validation: 10%, 25%, 50%, 75% and 90%. For example, if the penetration rate of Level 5 vehicles was 10%, then the other 90% was considered as CHV. Also note that Level 5 vehicles were all considered to have the most assertive driving behaviour, CAV B4, in the mixed-driving scenarios.

6.2.5. Traffic Control Methods

The AI traffic control method proposed in this research work was benchmarked against two control methods during the validation procedure: the fixed-time TLC and the FCFS (Dresner and Stone, 2008) heuristic rule-based control methods.

AI traffic control method:

The parameter set obtained after the training process for the AI model was frozen during validation, meaning the agent no longer executed backpropagation for further parameter optimisation. With this in mind, Critic, Actor Target and Critic Target networks were no longer required as the agent only operated in forward-pass mode during validation. In addition, the agent only exploited the knowledge and

experience it learned about the environment rather than exploring new policies for action selection. During training, stochasticity in the agent policy was introduced by adding a mean-zero Gaussian noise in the action selection. That meant that the agent did not always take the same action that gave good reward in a certain state, but it explored other actions to determine whether there were any other better actions that could give higher rewards. During validation, this exploration was not required, and the noise was removed in order to make decisions according to the optimal policy.

Fixed-time TLC method:

The fixed-time TLC method used in this work utilised the V2I communication interface to inform the approaching vehicles about the signal switch times so that they could set their speed profiles as they approached to the intersection accordingly based on the Wiedemann driving model in Vissim. In Vissim, the TLC model is provided by Vissig (PTV – Planung Transport Verkehr AG, 2019) add-on software module in which the control parameters for the model can be set via the Graphical User Interface (GUI). It is important to mention that the great majority of the traffic control systems do not have the V2I communication capability implemented as of today. The rationale behind considering this as part of the TLC method was to have a stronger benchmark control method that also considered a realistic near-future scenario.

The TLC method uses the concept of phases and stages. Phase refers to a group of traffic movement directions. For example, at a 4-way junction with 4 approaching links, there are 4 phases. Stage refers to a group of non-conflicting phases. The phase split and offset times were determined via an in-built optimisation process in Vissim. This process involved changing the green and red traffic light duration times for each phase iteratively until the best results were obtained in terms of the lowest average vehicle delay and the highest traffic flow. The aforementioned optimisation process steps are given in detail in Appendix A. The TLC method and the optimisation of its parameters represent the real-world deployment process of such control methods by traffic engineers.

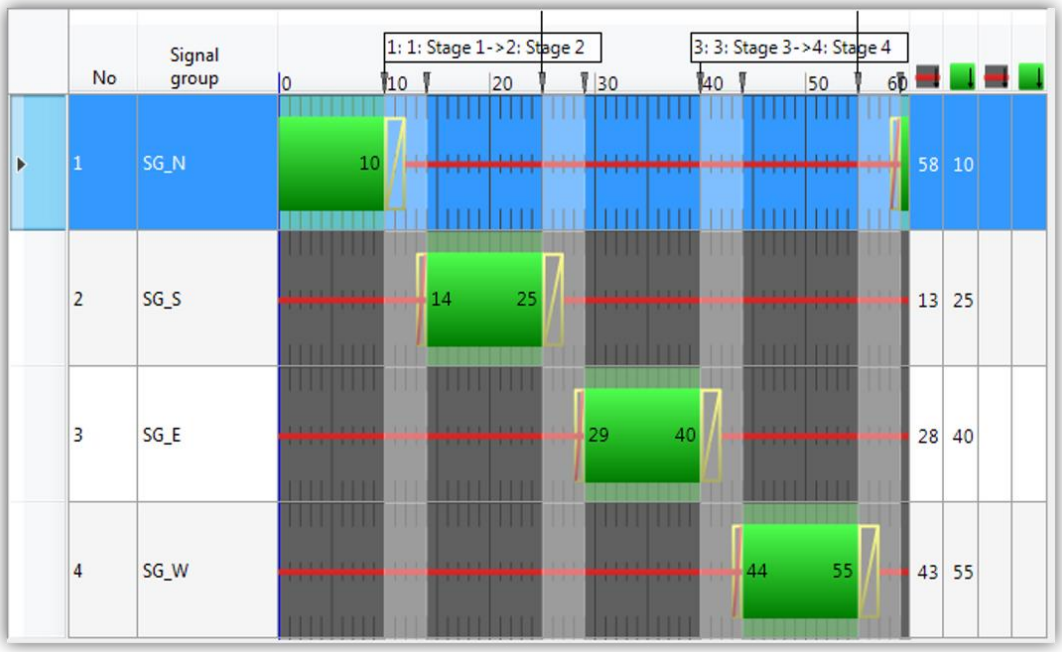


Figure 21 – Stage-based fixed-time TLC setup user interface in Vissim. Green light duration is shown with green tubes, red light duration is shown with red line and the switching times (amber/red light) are highlighted with yellow crossed box.

At the end of the optimisation procedure, the optimal phase and split times were found. The control cycle time was considered as 60 seconds and the optimised green light durations for each phase is shown in Figure 21. Each phase is defined as a group of traffic lights and the notations *SG_N, SG_S, SG_W* and *SG_E* are used that refer to Signal Group (SG) North, South, West and East for the road networks used in Figure 20 during validation.

FCFS control method:

The main difference between the FCFS and the AI methods is the way approaching vehicles are prioritised for intersection crossing. Essentially, the FCFS method gives priority to the vehicles based on their arrival time to the intersection without considering any other traffic condition i.e. queue length, vehicle delays etc. This method is widely used in the literature as a benchmark for unsignalised traffic control (Khayatian *et al.*, 2020).

6.2.6. Scenarios Overview

The overview of the generated validation scenarios is given in Figure 22. The scenarios are split into 2 sections, A and B. Section A represents the base validation scenarios whereas section B represents the key use cases in traffic. The combination of Section B use cases was used for validation on all base

scenarios in Section A. With this in mind, the total number of validation scenarios, n_vs_{total} can be found as:

$$n\_vs_{total} = \sum_{i=1}^{n=4} C(u_i, 1) * n\_vs_A \tag{23}$$

where n_vs_A is the total number of base scenarios in Section A, u_i is the number of use cases for each category from Section B and $C(u_i, 1)$ is the combination of those use cases within a category. This gives 360 validation scenarios in total.

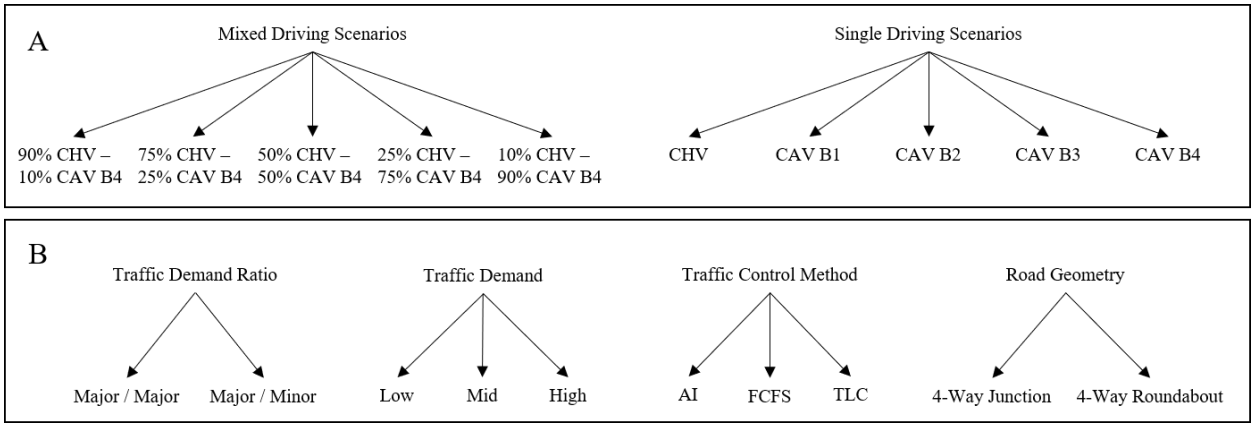


Figure 22 – Simulation test cases overview

The duration of each validation scenario was set as 2 hours and every scenario was run 5 times in order to capture the average behaviour among all simulation runs. In summary, a total number of 1800 (5 scenario repeats x 360 total scenarios) simulation runs were executed that added up to 3600 hours of traffic environment simulation. Random seed parameter in Vissim was varied for each simulation run to reflect the inherent uncertainty in traffic environment.

The results of each simulation run were automatically saved into a folder by Vissim that enabled performance analysis and comparison of the proposed unsignalised traffic control against the two benchmarked control methods.

6.3. Performance Metrics

The performance of the traffic control methods used during validation was measured and compared to each other based on a set of performance metrics. The identified metrics are associated with the environmental, congestion, travel time impacts of the control method under test on the traffic environment. In this research work, 8 performance metrics were defined in total; vehicle delay, number of vehicle stops, vehicle speed, queue length, fuel consumption and gas emissions. All performance metrics were calculated by Vissim during a simulation run and the results were logged into a file for data analysis afterwards. The details of these metrics are given in the following sub-sections.

6.3.1. Vehicle Delay

Vehicle delay time per vehicle in Vissim considers the additional time incurred due to congestion and the traffic control method by subtracting the theoretical attainable travel time from the actual travel time. Dividing the total delay of all vehicles in the network by the total number of vehicles in the network gives the average vehicle delay, and it was used as a performance metric. The same average vehicle delay was also used as one of the observation vector features as explained in Section 4.3.1.

6.3.2. Number of Vehicle Stops

A vehicle is considered as stopped when its speed in the previous timestep was greater than 0 (zero) and it is 0 (zero) in current timestep. Following this, the stop counter for that vehicle is incremented by one. Dividing the total number of stops of all vehicles by the total number of vehicles in the network gives the average number of vehicle stops, and it was used as a performance metric.

6.3.3. Vehicle Speed

Average vehicle speed is used as a performance metric, and it is also used in the state representation. Please refer to Section 4.3.1 for an explanation of its calculation.

6.3.4. Queue Length

Queue length is measured per lane from the upstream position of the queue, the CrA

entry point, up to the last vehicle that satisfies the “in-queue” condition which is determined as below:

- If the speed of a vehicle is less than 5 km/h, then that vehicle is considered as entering a queue.

- A vehicle remains in the queue as long as the speed of that vehicle has not yet exceeded 10 km/h.

Queue length is measured in Vissim in terms of units of length (i.e. metres), not in terms of number of vehicles. Average queue length was used as a performance metric, and it is calculated by measuring the queue length on each lane at each timestep and taking the arithmetic mean of the measured values, including 0 (Zero) values as queue length, for the duration of the validation scenario.

6.3.5. Fuel Consumption

All vehicles in Vissim were modelled as petrol vehicles and there was no other type of vehicle in the network. Fuel consumption was measured for each vehicle in terms of US liquid gram and the average fuel consumption was used as a performance metric.

6.3.6. Gas Emissions

Environmental impact assessment of the traffic control methods is done based on the measurement of exhaust emission of the vehicles in the network. The gas emission calculations in Vissim are based Traffic Network Study Tool Version 7F (TRANSYT7-F) simulation and optimisation tool by Penic and Upchurch (1992). Carbon monoxide (CO), nitrogen oxide (NOx) and volatile organic compounds (VOC) were the gas emission of interest for the validation scenarios. The average gas emissions for the duration of the validation scenario were used as performance metrics, and they were measured in terms of grams.

6.4. Summary

The validation procedure for the trained AI model have been explained in this section. In particular, the AI model is expected to generalise its optimal policy by applying the learned knowledge to previously unseen data during validation. To this end, 360 validation scenarios were generated that differ from the training scenarios in terms of road geometry (i.e. roundabout), mixed-driving and traffic demand ratios. In addition to this, benchmark traffic control methods were also tested under the same scenarios in order to compare their performance in terms of the identified metrics.

The validation traffic scenarios were designed to isolate the impacts of particular traffic control methods on traffic flow, journey times and congestion in the presence of CAVs. Whilst they did not represent real world situations exactly, due to the assumptions made in this research work, the validation results gave an indication of relative performance improvements in the traffic environment.

The computer simulation validation results and the discussions are presented in Chapter 8 together with the validation results of the scaled real-world experiments. The next chapter will focus on the development of the scaled road network with scaled CAVs for the validation of the unsignalised traffic control on the scaled testbed.

Chapter 7

7. Validation of AI Traffic Control in Scaled Testbed Experiments

7.1. Introduction

A simulation based validation process can offer great benefits in terms of avoiding costly installations with physical assets, speeding up experiments, varying conditions and scenarios easily and generating edge cases for safety-critical situations. On the other hand, practical testing and validation is also not avoidable mainly due to the fact that a thorough understanding of a traffic environment for modelling in simulation including all factors involved is very challenging, if not impossible. Therefore, simulation work can accelerate the validation and testing activities of complex systems such as traffic control in real-world, but not necessarily replace these activities.

Taking that into consideration, a scaled testbed setup with scaled CAVs is presented in this chapter for executing validation scenarios with physical assets and measuring performance of the proposed traffic control system. Although the road network and the CAVs are scaled, the testbed can be seen as a bridge between a simulation work and a full-scale deployment in a real-world setting. Indoor localisation, V2I communication, traffic monitoring and visualisation capabilities are some of the key features implemented. Unlike the scaled testbed setup by Stager et al. (2017), all scaled cars in this work have the capability to run vehicle behaviour, decision making and motion control algorithms on-board without requiring an external processing unit. In addition, a digital twin of the scaled testbed was also created in Vissim to cross-validate the vehicle behaviour and the traffic control performance, and is presented in the following sections.

7.2. Scaled Testbed Environment

7.2.1. Road Network Setup

The scaled road network in this research work was used as a testbed to further validate the traffic scenarios in a controlled environment. It incorporated realistic cues with regards to the traffic

environment including a 2-way intersection with a single lane on each link, scaled CAVs running on AI-enabled embedded systems, V2I wireless communications based on Wi-Fi, vehicle positioning and localisation system.

Different versions of the road network have been used during the initial stages of the project in an iterative manner until all required traffic environment features and functions were implemented. Figure 23 shows three versions of the road network in which the top-left picture is the Version 1 (V1), and the bottom picture is the Version 3 (V3) that is the final version used for the validation scenarios.

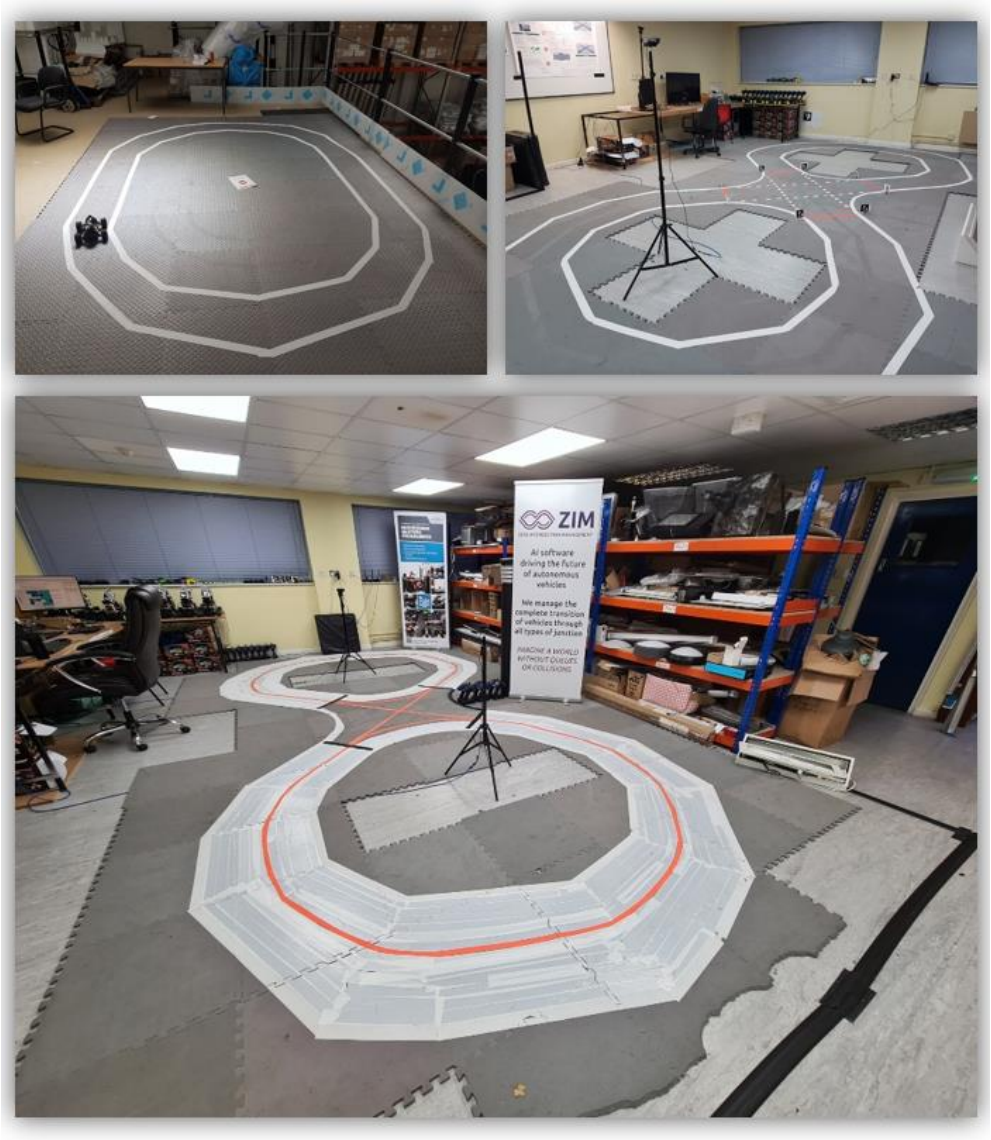


Figure 23 – Scaled road network setup for unsignalised traffic control experiments. Version 1 (top-left), Version 2 (top-right) and Version 3 (bottom) of the road network setup are shown.

The final version of the road network spanned over 30.24 square metres (7.2m x 4.2m), and the road surface was constructed out of 84 interlocking Ethylene-Vinyl Acetate (EVA) foams (0.6m x 0.6m) with 10 mm thickness. The road lanes were built with white matt gaffer tape, and the middle of the road was marked with orange tape. The intersection entry and exit points were marked with black tapes. The initial experiments on V1 and V2 showed that marking only the outer boundaries of the lanes made it very challenging for the scaled cars to determine which side of the white lane boundary to drive on the curved sections of the road due to the on-board car camera not being able to see both lane boundaries at the same time. This problem was resolved by marking the whole road lane with white colour in V3.

Figure 24 – Demonstration of the unsignalised control method at an event in the UK

Having a portable road network also facilitated the demonstration of the traffic control method at various different academic and industrial events during the project. Figure 24 shows the road network setup process at one of those events in the UK, it took 30 minutes for one person to complete the full system setup for the demonstration. The complete list of the Bill of Materials (BOM) for the scaled testbed is given in Appendix B.

7.2.2. Wireless Communications

The V2I communication between the scaled cars and the TCA was established via Netgear Nighthawk Smart Wi-Fi Router AC1900. The router supported IEEE 802.11 b/g/n variants at 2.4 GHz and IEEE 802.11 a/n/ac variants at 5.0 GHz. In this project, 2.4 GHz range was used for communication. The TCA was run on a Windows 10 machine acting as the RSU on the scaled testbed.

7.2.3. Indoor Localisation and Positioning

Vehicle localisation is one of the key requirements to satisfy in order to determine the distance of the scaled cars to the intersection entry point. GPS based localisation would fail to work or lack precision

required indoors. Therefore, the adopted method in this research work was based on external cameras and ArUco markers (Garrido-Jurado *et al.*, 2014) that were placed on top of the scaled cars. There were 2 external cameras (Logitech C922 Pro Stream) on tripods as shown in Figure 24 that captured the whole testbed when combined.

ArUco marker was composed of a white binary matrix on a black background. In this work, the marker size of 4x4 was used in terms of the number of bits per marker. The ArUco marker concept is originated from the pose estimation requirements of robotic applications, and the concept enables 3D translation of position and rotation vectors from the binary matrix of the markers to be obtained. The software implementation of this concept was based on OpenCV library (Itseez, 2015) in which marker detection, identification and pose estimation functions were all defined and provided as a library in Python programming language. The ArUco markers that were used on the scaled cars are shown in Figure 25 with their embedded IDs written underneath the markers.

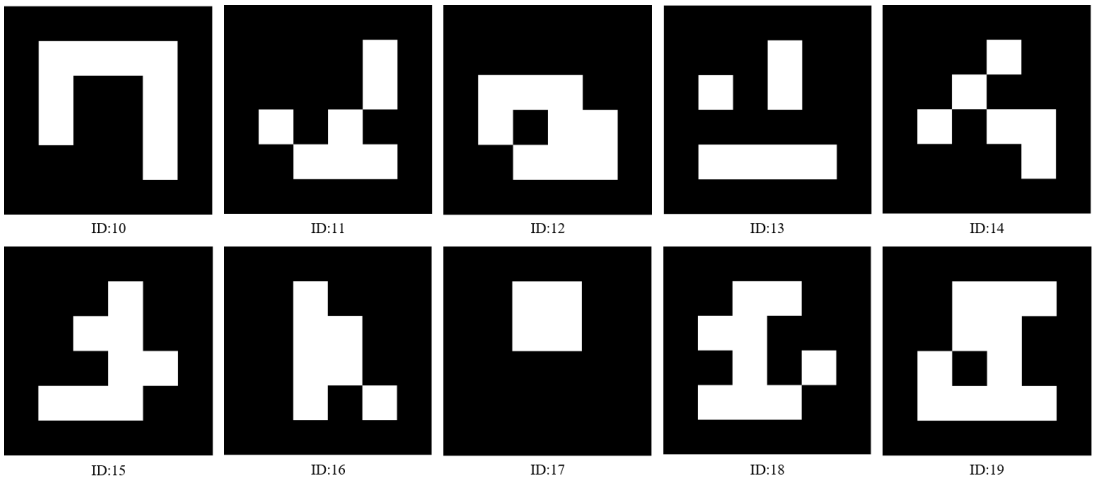


Figure 25 – Unique ArUco markers for the scaled car identification

The ArUco markers were placed on top of the scaled cars as the screen capture of the external cameras show in Figure 26. A cube-shaped paper with a triangle roof was used to display the ArUco marker on every surface to ensure visibility from all locations on the testbed. The indoor localisation and positioning software function run every 25 ms. Firstly, it detected all the markers on the testbed and identified each car. Secondly, it calculated the direction and location of the cars in terms of distance to the intersection entry points. Finally, the latest localisation data was sent over to the vehicles via V2I communication interface. The accuracy of the localisation was measured to be better than 1 cm.

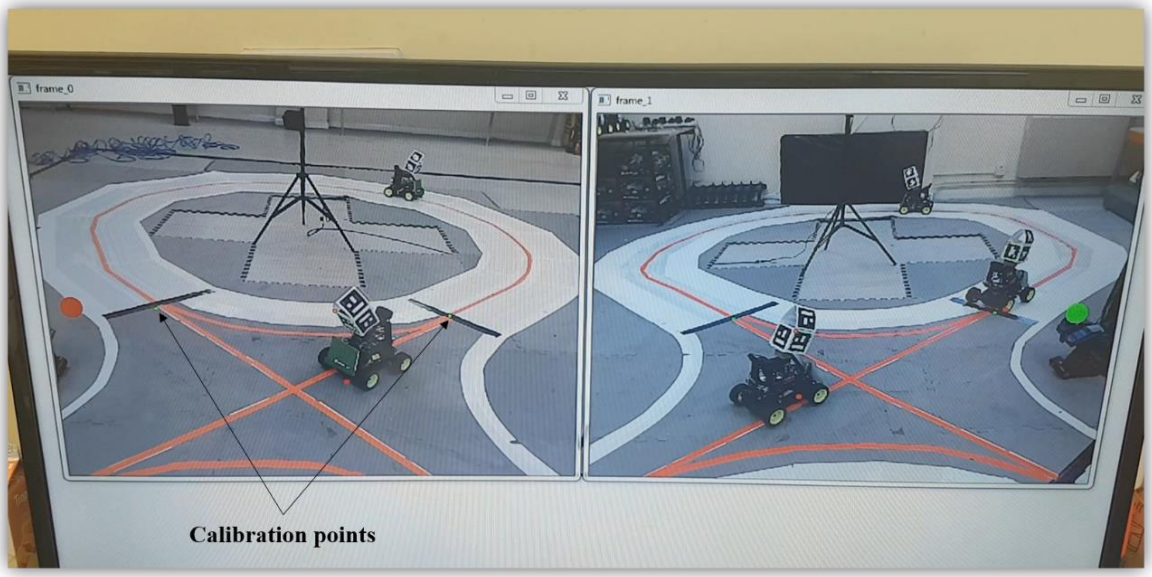


Figure 26 – Indoor localisation and positioning with Aruco markers. The laptop screen shows side by side what both cameras capture. The shared intersection space is captured by both cameras whereas the curved sections of the road network are only captured by one camera.

When the testbed is moved to another location for demonstration purposes, it is important to position the cameras correctly. To this end, a calibration procedure was implemented to accelerate the setup of the cameras in the correct position. The green dots on the intersection entry and exit lines in Figure 26 are essentially the calibration points which must overlay on top of the intersecting point where black and orange lines meet.

7.2.4. Digital Twin

A digital twin is defined as the relevant abstraction of a physical system, rather than being an exact replica of that system, including modelling of complex behaviours and interactions. In this project, the digital twin of the scaled testbed was created in Vissim in order to cross validate the scaled testbed experiment results and to calibrate the driving behaviour parameters on the scaled cars to achieve uniform behaviour in both platforms.

Figure 27 – The digital twin (left-image) of the scaled testbed (right-image) in Vissim.

The length of a passenger car model in Vissim was similar to a real size passenger car i.e. 4.5 m. However, the length of the scaled cars was 0.25 m. This necessitated a scaling factor to be applied to the road network as well in order to achieve similar performance results in both platforms. The radius of each loop on the figure of eight scaled road network was 2.5 m which is 10 times greater than the length of the scaled car. With this in mind, the radius on the digital twin was also set as 10 times of a passenger car length i.e. 45 m.

In computer simulations, 5 different driving behaviours (1 CHV, 4 CAV B1 to B4) were generated by configuring the driving model parameters within Vissim. However, implementing these driving behaviours in the scaled cars was challenging due to the Vissim driving model not being available in the scaled cars. Therefore, the driving behaviour in the scaled car was calibrated to behave similar to CAV B1 driving behaviour in the simulation tool by utilising the digital twin setup. The calibration process involved iterative 15-min experiments on the scaled testbed with different sets of driving behaviour parameters, which provided the intersection throughput values in terms of the total number of vehicles that crossed the intersection, and choosing the parameter set that gave the nearest results when compared to the digital twin simulation experiments.

The reason for choosing CAV B1 driving behaviour only was mainly due to the limitations of the sensors on the scaled cars which did not make it possible to follow the preceding car on the scaled testbed with reduced headways. The validation scenarios that will be presented in the next section were executed both in the scaled testbed and in the digital twin. The digital twin simulations did not run in parallel or real-time as the scaled testbed experiments. The experiment results were obtained independent from each other.

7.2.5. Validation Scenarios

The validation scenarios on the scaled testbed mainly focused on the comparison of the TLC and the AI control methods under varying traffic demand conditions. It is important to mention that the variety of scenarios that can be implemented on the scaled testbed is much more limited compared to the simulation work in terms of the number of vehicles. There were 10 scaled cars in total built for the purpose of these experiments. However, maximum 6 of these cars could run at the same time due to the size restrictions of the testbed.

In summary, the validation scenarios were a combination of traffic control methods and the traffic demands as shown in Figure 28. Each square box represents one unique scenario in which the number of scaled cars to run is listed on top and the demand ratio, in terms of Right-Turn (RT), Straight (S) or Left-Turn (LT) at the intersection crossing, is given as the distribution of the scaled cars for each potential route. This gives a total of 16 validation scenarios. The combination of LT, S and RT routes were used in order to vary the vehicle trajectory choices during intersection crossing.

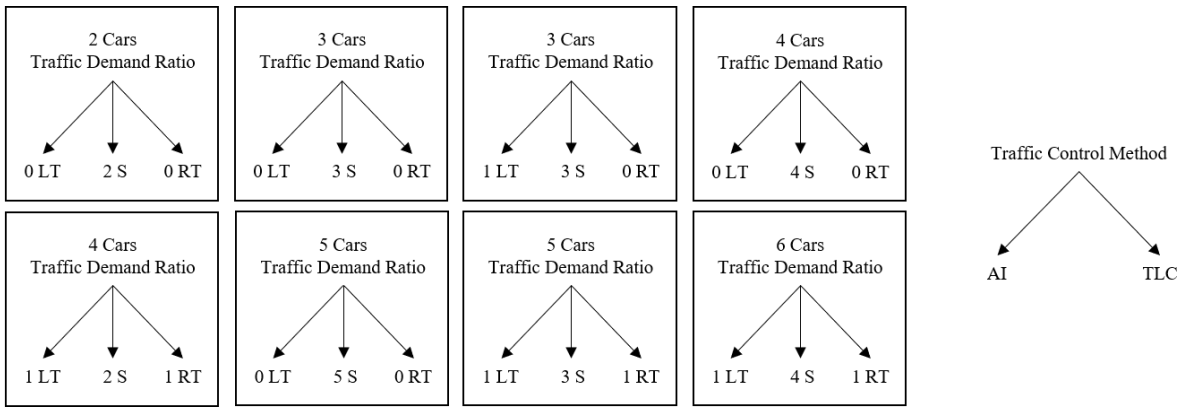


Figure 28 – The scaled testbed scenarios overview

Each validation scenario was run for 15 minutes, and it was repeated twice both in the scaled testbed and in the digital twin. The experiment results that will be presented in Chapter 8 are the average value of these multiple runs for each scenario. Two performance metrics were defined to quantify the performance of the traffic control methods; average vehicle delay and the intersection throughput which is the total number of vehicles that cross the intersection. Some of the performance metrics used during the simulation work could not be used in the scaled testbed i.e. fuel consumption, gas emissions as they depended on the algorithms and models implemented in Vissim.

7.3. Scaled CAVs

7.3.1. Vehicle Hardware Components

The key hardware components of a scaled CAV are explained in this section. A total of 10 scaled cars have been built to the exact same standards. In a nutshell, each car has the ability to run complex AI algorithms in real-time, capture images for driving, communicate wirelessly with the RSU, detect objects, measure speed, control steering and throttle.

Chassis, Motor and Batteries:

The scaled CAVs were based on an off-the-shelf Remote-Controlled (RC) car at 1/16 scale with a brushed motor and an Electronic Speed Control (ESC) unit as shown in Figure 29. The chassis was large enough to carry all required components, and it was also small enough to operate in the scaled testbed area. The original RC car 1100 mAh 7.2V Nickel–Cadmium (Ni-Cd) battery was replaced with a 4200 mAh 7.2V Nickel–Metal Hydride (NiMH) battery with increased battery capacity for longer operation time. When fully charged, a scaled CAV could drive for about 45 minutes continuously. A charging station was setup near the experiment area to manage battery charging under safe conditions by using fire and explosion proof safety bags.

Figure 29 – The original remote-controlled car that was purchased off-the-shelf (top-left-image), fleet of cars during the electronics assembly process (right-image) and the final version of the assembled car (bottom-left-image) are shown.

During the initial experiments with the scaled CAVs, a performance comparison was carried out between a brushed motor and a brushless motor to determine the suitability of these options for the project. The experiments revealed that the speed control with a brushless motor was very difficult at low speeds (under 5 m/s). Therefore, a brushed motor was used in all cars. The scaled cars could go up to 50 km/h speed, and the chassis was equipped with axle trees to reduce friction of the components and to run smoother. In addition to that, there were front/rear bumpers and spring shock units that provide protection against impacts.

The brushed motor was controlled with Pulse Width Modulation (PWM) signal, and for that reason, a dedicated 16-Channel 12-bit PWM driver board was used supplied by Adafruit Industries. The PWM driver acted as a bridge between the computing platform and the brushed motor that converted the digital control requests into analogue signals.

Computing Platform:

An onboard computing platform was integrated into the vehicle that handled the automated driving operations in the scaled testbed. A trade-off was made between cost and computing speed in terms of frames per second (fps) by deciding on a minimum viable fps of 20. This was determined based on the maximum speed limit of 2 m/s set for the scaled CAVs. In other words, a scaled car would travel maximum 5 cm between two consecutive frames which was deemed as satisfactory. With this in mind, Raspberry Pi 3 Model B+ and NVIDIA Jetson Nano Developer Kit platforms were selected for the initial experiments, and the two cars in Figure 30 were built to the same specification apart from the computing platform. At the time of writing this thesis, the cost of Raspberry Pi solution was three times less than that of NVIDIA solution. However, the performance comparison testing showed that the Raspberry Pi board processing speed was 12 fps on average whereas it was 96 fps for the NVIDIA board. As a result, NVIDIA solution was selected as the main computing platform for the project.

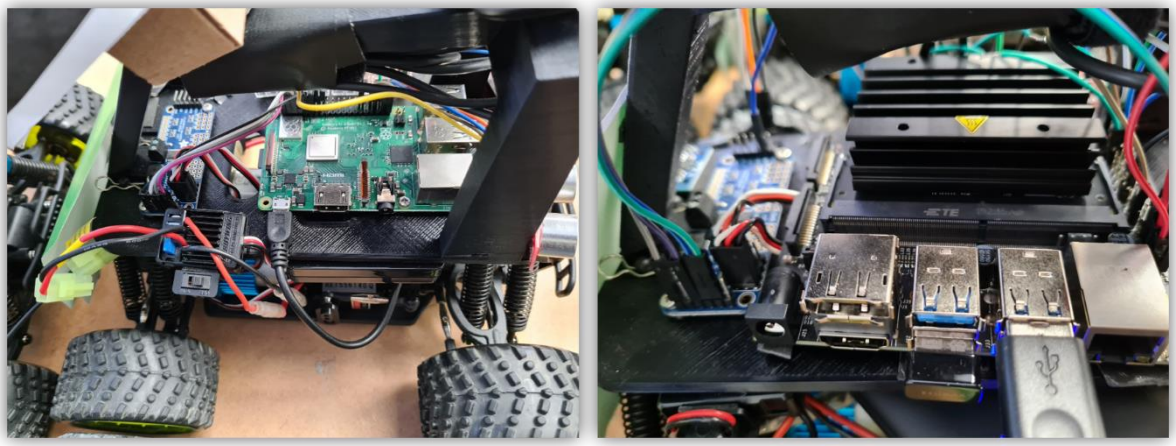


Figure 30 – Comparison of the scaled CAV design based on Raspberry Pi 3 Model B+ (left-image) and NVIDIA Jetson Nano (right image)

The computing platform and other electronic components were not powered by the same battery as the motor control. The secondary battery was a 10 Ah lithium-ion polymer battery with 3A maximum current capability. All electronic components were housed on a 3D printed black assembly which was made out of polylactic acid (PLA) material, and the design was based on the open-source donkeycar platform (Donkeycar, 2021).

The NVIDIA board did not include any wireless connectivity Integrated Circuit (IC) unlike the Raspberry Pi solution. Therefore, the Edimax N150 USB dongle that provided Bluetooth and Wi-Fi connectivity was used on all cars for V2I communications.

Camera:

The on-board camera is one of the key components that capture the traffic environment as an image input for the automated driving AI model. Logitech HD Webcam C525 was selected due to its compact size, ease of integration with the computing platform via USB interface and low cost. The resolution in pixels was not a key requirement as down-sampling was performed on the camera image prior to feeding it into the neural network.

Sensors:

All vehicles were fitted with some additional sensors which were controlled and monitored by the main computing platform. HC-SR04 ultrasonic sensor from SparkFun Electronics was one of the sensors integrated for object detection. Two ultrasonic sensors were placed at 45 degree angle on the front left

and front right sides of the scaled cars. The validation scenarios did not require the cars to reverse at any moment in time. Hence, no sensor was placed on the back side of the cars. The ultrasonic sensor provided 2-400 cm of non-contact distance measurement functionality from the object with an accuracy of no worse than 10 mm.

Figure 31 – Speed measurement of the CAVs with hall effect sensors

Speed measurement was achieved by a hall-effect sensor that was installed inside of the back wheels of the scaled cars as shown in Figure 31. The sensor produced a trigger signal when the magnetic field density around it exceeded a set threshold value, and this was captured by the computing platform as a digital input signal. Magnetic discs (6 in total) with 5 mm radius were placed inside one of the wheels at an equal distance from each other. By doing this, the distance travelled by a scaled car could be found based on the fact that one revolution of the wheel was 24 cm, and 6 trigger signals were generated per revolution. Then, the average speed of a vehicle within a monitoring cycle of 200 ms could be found by solving the equation, distance = speed * time.

An Inertial Measurement Unit (IMU) based speed measurement was also considered and tested on the scaled cars. The MPU6050 sensor from InvenSense Inc. was used due to its low cost and compatibility with the computing platform. The sensor had 3-axis gyroscope and a 3-axis accelerometer for directional speed measurement. However, the scaled vehicle testing showed that the accuracy was very poor, reaching up to 0.5m, and the error accumulated if the sensor was not calibrated continuously. Therefore, an IMU based speed measurement method was not chosen as a viable solution in this project.

This concludes the key hardware components. The complete list of BOM for the scaled cars is given in Appendix B with the associated cost information.

7.3.2. Vehicle Software Components

The vehicle software in this project refers to the automated driving application that runs on the computing platform, and the software architecture was inspired by the open-source donkeycar software library (Donkeycar, 2021). The software was executed in a multi-threaded way where key tasks run in parallel at different cycle rates. The aforementioned key tasks are shown in Figure 32 with their cycle rates listed underneath, ranging from 20 Hz to 100 Hz. The arrows indicate the producer-consumer relationship between the tasks in which a producer generates the required data to be used by the consumer task for decision making or post-processing.

The main system task run at 100 Hz and responsible for starting, terminating and monitoring other tasks in the software in addition to handling data exchange between these tasks when required. Object detection and camera vision tasks had the next fastest cycle rates as they had the greatest impact on driving within the lanes and preventing collisions with other cars. All vehicle software components run on Linux Operating System (OS), and they were coded in Python programming language.

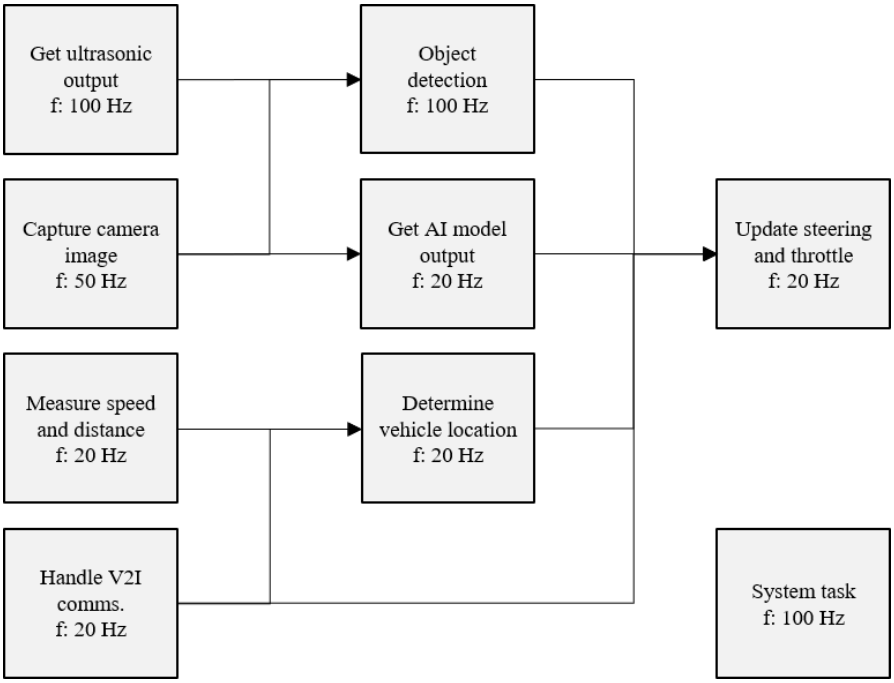


Figure 32 – The scaled CAV key software modules for automated driving application that runs on the computing platform.

7.3.3. Automated Driving AI Model Setup

The neural network for the automated driving AI model is given in Figure 33. The network consists of an input layer, 6 Convolutional Layers (CL), 5 Fully-Connected (FC) layers and 2 output layers with their corresponding number of neurons as shown in the figure.

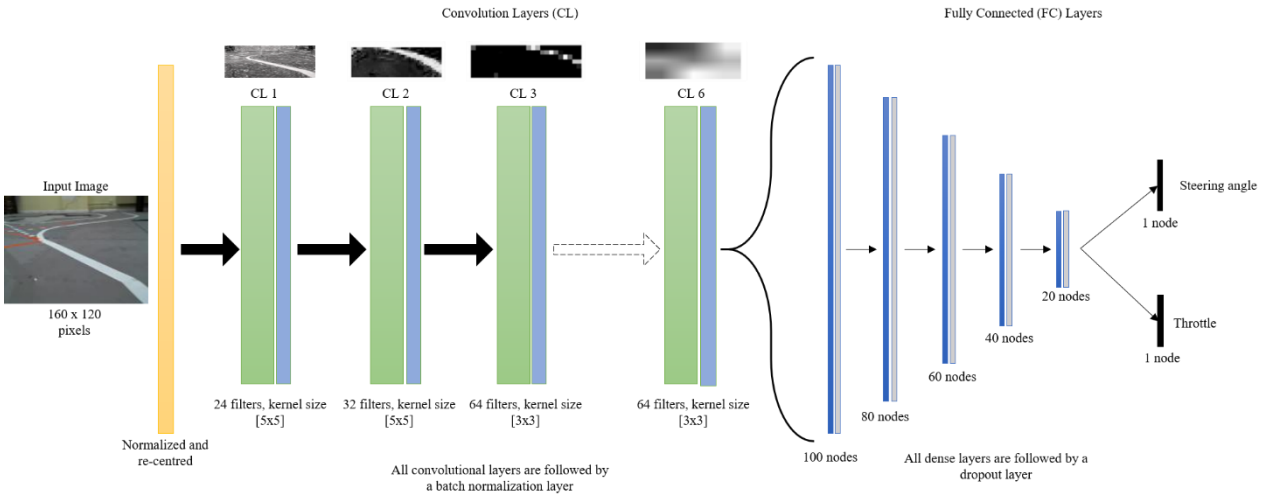


Figure 33 – AI network for the automated driving task

The input layer takes the down-sampled (160x120 pixels) camera image as an input and performs normalisation on the pixel values in the target range of [0, 255]. The CLs perform feature extraction, and the process of an input image being abstracted to a feature map is shown on top of the CLs in Figure 33. The sequence of images helps to visualise what happens to the original input image after each CL. The CLs are followed with FC layers leading to 2 output layers for throttle and steering control values which are produced in the range of [-1, 1] by the network, and then, they are scaled accordingly afterwards for actual steering and throttle command to the electric motor.

The neural network architecture can be considered as an end-to-end model, similar to Bojarski *et al.* (2016), based on the fact that the network parameters are optimised by considering the input and output data directly without including any intermediate data points. The next section will present the training procedure for the AI model.

7.3.4. Automated Driving AI Model Training

The training method for the automated driving AI model is based on supervised learning. This means that the training data is collected and labelled offline unlike the RL training where an agent is trained online by interacting with an environment. In the context of automated driving with scaled cars, the objective of the training for the agent is to predict throttle and steering angle control values for the given input image based on the labelled data acting as the ground truth.

The training data was collected in the scaled testbed by driving one of the scaled cars with a remote control device. In other words, the ground truth values came from a human driver as the scaled car was controlled remotely. It was important to control the scaled car as smoothly as possible without going outside the road lanes or changing the speed frequently. Because the agent was trained on this collected data to behave as close to the human driver as possible.

Data collection was done over 20 minutes of driving the scaled car remotely around the testbed per each potential route at the intersection crossing. There were 3 potential routes, LT, S and RT. As a result, 60 minutes of driving data was collected which corresponds to about 60k labelled records. A single record refers to a collection of data points that include the camera image, timestamp, throttle, steering angle, distance from the intersection entry and the route choice if crossing the intersection. An example set of data collected during training is shown in Figure 34. In this figure, the images on the left and right hand sides correspond to driving within the lane and crossing the intersection. As can be seen, crossing an

intersection can be challenging for an AI model due to the complexity of the road markings. This is why the training data was collected for each route option so that the agent could be taught how to turn left, right or go straight accordingly.

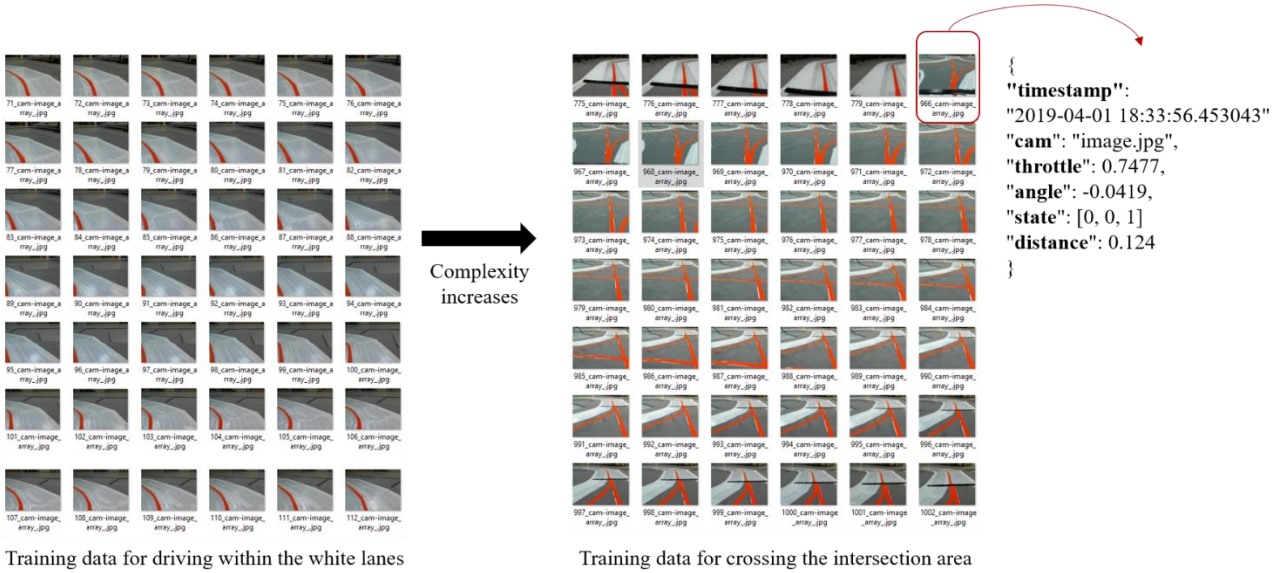


Figure 34 – Training data collection for automated driving AI model within the scaled road network

Training data collection is a lengthy process during which the human driver may not always be able to control the scaled car perfectly in the middle of the road. However, this does not pose a serious problem as long as it is only a small portion of the collected data.

Generalisation of automated driving behaviour to unseen situations during training is very important. To this end, data augmentation was applied to the training data. Data augmentation refers to the process of applying little variations to the original image in order to multiply the amount of training data in a synthetic way. This technique reduces overfitting of the neural network parameters, and thus makes the AI model to be more robust in the automated driving task. Figure 35 shows an original image on the far left and the 3 synthetic versions of it from left to right where the image attributes, in terms of brightness contrast and gamma correction, are varied. In summary, 60k original training data was increased to 240k in this project by applying data augmentation on each image.

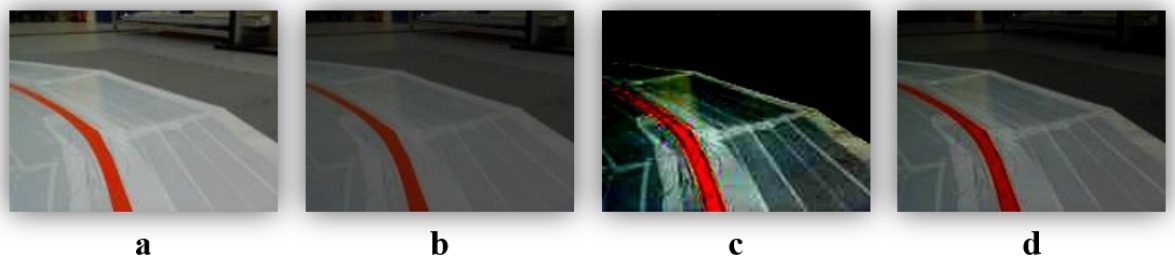


Figure 35 – Data augmentation for improving the quality of the training data. a) Original image, b) Brightness contrast filter applied, c) night vision filter applied, d) gamma correction filter applied

7.4. Summary

The testbed introduced in this chapter is a scaled road network in a figure of eight shape with a single road lane. It has been constructed with an objective of replicating real-world traffic control scenarios in a scaled, cost-effective and controlled environment. It can be seen as a bridge between a simulation work and a real-world deployment of such a system. Realistic environmental cues have been included in the scaled testbed such as V2I communications, road markings and intersection crossing shared space so that the impact of an unsignalised traffic control could be obtained and compared with a traffic light based control method. Furthermore, the digital twin of the scaled testbed has also been created in Vissim to cross validate the experiment results.

The technical details of the scaled CAVs are also presented in this chapter. During the development of the scaled cars, some methods or approaches did not work and all of these important "lesson-learned" points are also presented. The NVIDIA Jetson Nano processor was used as the main computing platform which run the software for the automated driving application. The automated driving task itself required a training data collection process with a human driver controlling one car remotely around the testbed. 10 cars have been assembled that were built to the same specification, and they could drive autonomously around the scaled testbed simultaneously.

The validation cases presented in this section have been executed in the scaled testbed and the results are presented in Chapter 8 together with a discussion of the key findings.

Chapter 8

8. Performance Evaluation of AI Traffic Control

8.1. Introduction

This chapter presents the results of the simulation work from Chapter 6 and the scaled testbed experiments from Chapter 7. The discussions focus on the impacts of the traffic control methods on traffic flow, congestion, journey times and the environment. The measure of effectiveness of the traffic control methods are quantified with metrics and the rest of this chapter is organised to present the key results based on these performance metrics. The remaining results in the form of data tables and graphs can be found in Appendix C.

8.2. Traffic Simulation Results

8.2.1. Vehicle Delay

The average vehicle delay times for the validation scenarios is shown in Figure 36. In this figure, the top and bottom three charts are for the scenarios where the demand ratio is major/major and major/minor respectively. The x-axis of all charts has a series of discrete points that represent five different driving behaviours and five different CAV penetration rates in mixed-fleet operation conditions (See Section 6.2). For example, 90% CHV label is used to represent a mixed-fleet traffic flow condition where 90% of the vehicles are CHV and 10% CAV B4.

Simulation data in Figure 36 shows that vehicle delay times decrease as traffic demand decreases from high to low in common for all traffic control methods as expected. Another common trend for all control methods was that the delay times dropped as CAV penetration ratio increased, and as vehicle driving behaviour became more assertive from B1 to B4. This trend was more pronounced when the traffic demand was high and the FCFS method was used. The FCFS method had the highest delay times in high traffic demand scenarios compared to the AI and the TLC methods, reaching up to 88.34 sec and 270.23 sec under major/major and major/minor demand ratios respectively. The AI method was shown to perform the best in all high demand traffic scenarios with minimum delay times of 12.26 sec and

12.67 sec under major/major and major/minor demand ratios respectively. When the traffic demand was medium or low, the TLC method displayed the worst performance in terms of vehicle delay time which was in the range of 20.52-21.96 sec while the AI and the FCFS methods were on a par in all scenarios with a maximum vehicle delay difference of 2 sec between the two methods.

Interestingly, the vehicle delay time for CAV B1 was observed to be greater than CHV by 8.36 sec when the demand ratio was major/major in high traffic demand scenario under only the FCFS method. A note of caution is due here since this difference in vehicle delay between CAV B1 and CHV was not found when the demand ratio was major/minor. A possible explanation for this can be the fact that CAV B1 is a more cautious driving behaviour than CHV and the effects of this is more pronounced under high traffic demand scenarios when two major roads intersect.

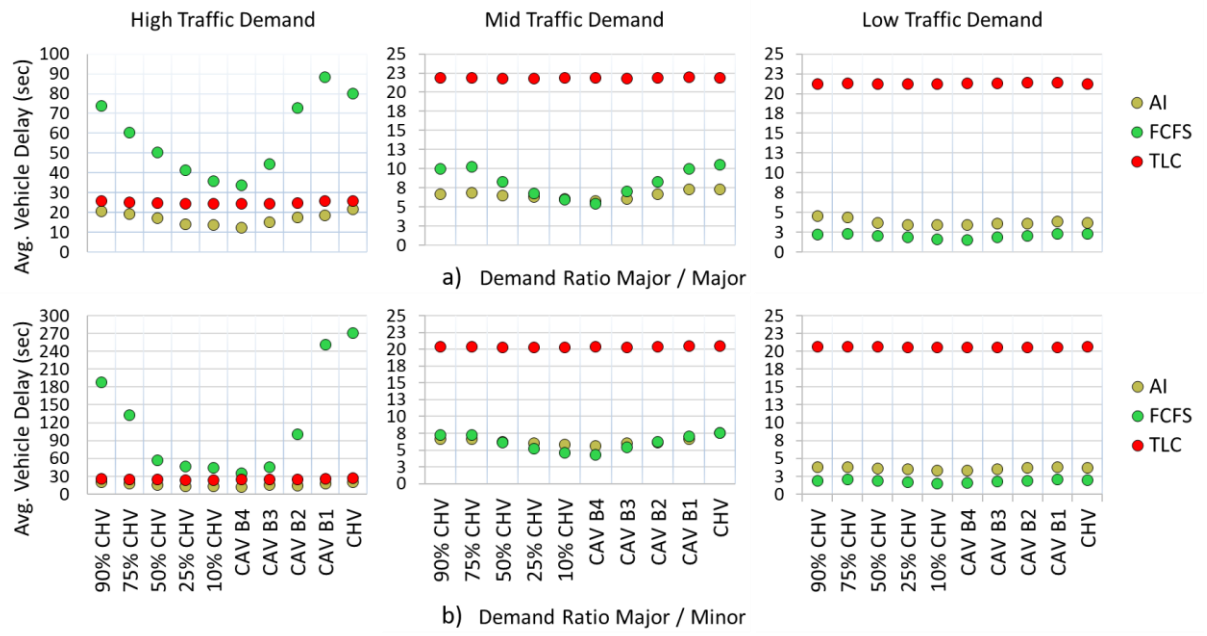


Figure 36 – Average vehicle delay for each driving behaviour and CAV penetration ratio under high, medium and low traffic demand scenarios a) when the demand ratio is major / major (top 3 graphs) and b) when the demand ratio is major / minor (bottom 3 graphs)

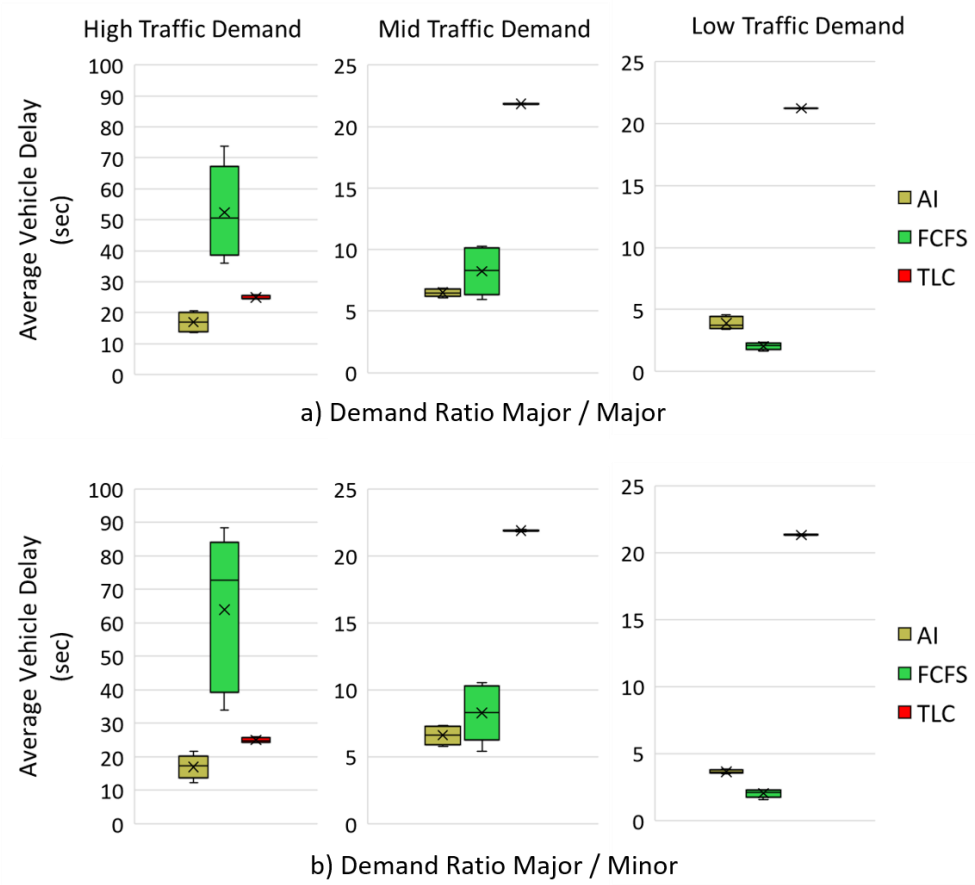


Figure 37 – Average vehicle delay box plot that shows the range of delay times for all driving behaviours and CAV penetration rates under each traffic control method. The middle line of the boxes and the x inside the boxes represent the median and mean values respectively. 1st quartile (bottom line of the boxes) and 3rd quartile (top line of the boxes) of the range are also shown together with the whiskers that represent the maximum (top) and minimum (bottom) values in the range a) when the demand ratio is major / major (top 3 graphs) and b) when the demand ratio is major / minor (bottom 3 graphs).

The range of vehicle delay values that were obtained from all scenarios of driving behaviour and CAV penetration ratio are shown in a box plot in Figure 37 for each traffic control method. From this figure, it can be seen that the TLC method has the most constrained range of delay times under all traffic demand conditions with the largest delay range of 24.44-25.96 sec under high traffic demand. This indicates that the CAV penetration rate and driving behaviour being assertive or cautious have minimum impact on the delay times with the TLC control. On the other hand, the FCFS method was shown to be very sensitive to changes in driving behaviour of vehicles and CAV penetration rate in the traffic flow as this control method had the widest range of delay times under all traffic demand conditions with a maximum delay range of 33.84-88.34 sec under high traffic demand. What stands out in this figure is that the AI method performance gets significantly better as traffic demand increases compared to the TLC method.

The average vehicle delay percentage improvement in all scenarios against the baseline scenario of CHV where the traffic flow consists of 100% CHV driving behaviour is shown in Figure 38. This figure is important mainly because it makes a performance comparison between the potential future traffic conditions and the traffic conditions of today where CAVs do not exist on a mass scale. What is striking in this figure is that a maximum of 10% improvement in vehicle delays could be obtained when the TLC method was used even if 90% of the traffic flow was CAV B4. On the other hand, the AI and the FCFS methods offered much greater improvements as the driving behaviour became more assertive or CAV penetration rate became higher.

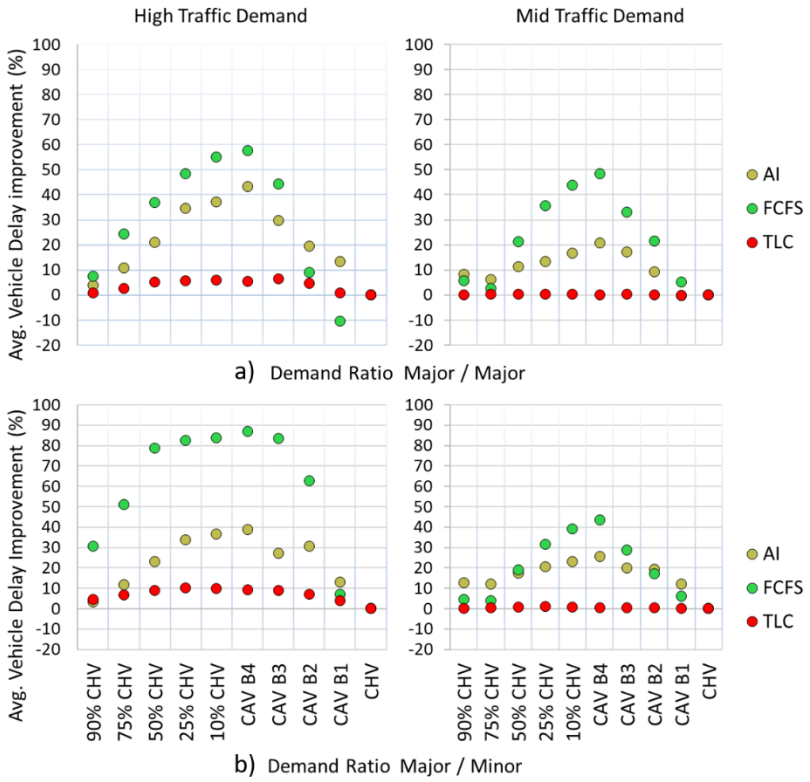


Figure 38 – Average vehicle delay percentage improvements of all scenarios are shown against the CHV scenario where the traffic flow consists of 100% CHV driving behaviour a) when the demand ratio is major / major (top 2 graphs) and b) when the demand ratio is major / minor (bottom 2 graphs).

During training of the AI method, as explained in Chapter 5, only a 4-way junction geometry was used as the road network. However, validation scenarios included a roundabout geometry as well which has not been used during training in order to validate the operation of AI method in an unseen geometric road network. Figure 39 shows the comparison of average vehicle times for 4-way junction and 4-way roundabout under major/major traffic demand ratio scenarios when AI method is used. It can be seen from this figure that the trend in vehicle delay under all scenarios are the same for both road networks

where the delay times are slightly higher in roundabout with a maximum difference of 3.69 sec between two road networks in high traffic demand scenarios. It is apparent from this figure that the AI method generalises well to other road networks that are not used during the training session.

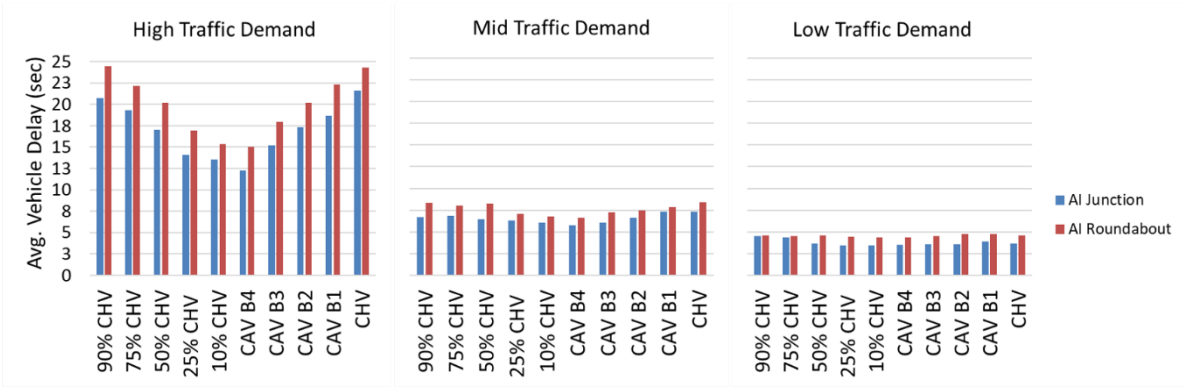


Figure 39 – Comparison of average vehicle delay for each driving behaviour and CAV penetration ratio 4-way junction and 4-way roundabout scenarios when the demand ratio is major / major.

8.2.2. Number of Vehicle Stops

In an ideal intersection crossing scenario, there should be no stop-and-go movement in order to incur no vehicle delays. Therefore, the number of vehicle stops metric gives useful insights with regards to congestion build-up. Figure 40 provides the results obtained from the validation scenarios in terms of the number of vehicle stops for each control method.

As can be seen in Figure 40, the number of vehicle stops reduces as more CAVs penetrate into the traffic flow or as driving behaviour gets more assertive with the AI and the FCFS methods. However, a closer inspection of the figure demonstrates that this trend does not hold true for the TLC method, and the greatest number of vehicle stops (1.03, 1.19, 1.35 under high, mid and low traffic demands respectively) was observed with CAV B4 driving behaviour. With the TLC method, the number of vehicle stops decreased as the CAV penetration rate reduced or driving behaviour became more cautious. In addition to that, the number of vehicle stops increased as traffic demand decreased with the TLC method which was the other way around with the AI and the FCFS methods. This result may be explained by the fact that the TLC method gives right-of-way to all approaching links in turn in a control cycle even though there are no vehicles waiting to cross in the queue on a particular approach link which becomes the case as traffic demand decreases.

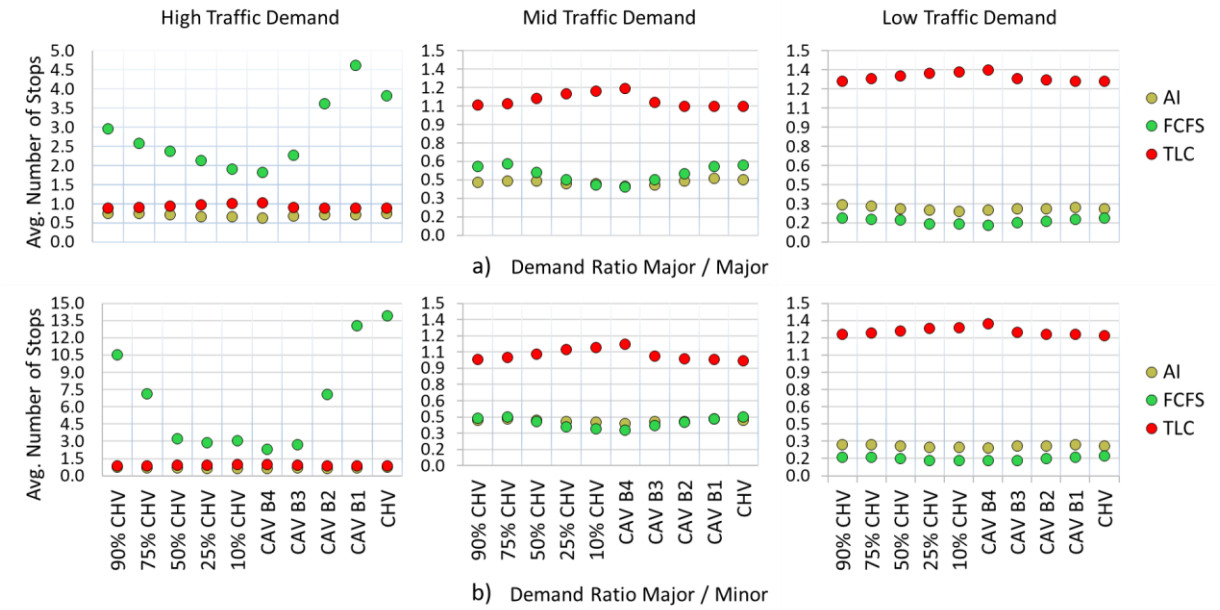


Figure 40 – Average number of vehicle stops for each driving behaviour and CAV penetration ratio under high, medium and low traffic demand scenarios a) when the demand ratio is major / major (top 3 graphs) and b) when the demand ratio is major / minor (bottom 3 graphs).

There were no control methods that eliminated the stop-and-go movement for the given traffic demand scenarios. However, the AI method specifically was shown to perform the best in majority of the scenarios except for the low traffic demand scenarios where the FCFS method outperformed the AI method marginally with a maximum difference of 0.1 average number of vehicle stops. It is also important to highlight that data from Figure 36 can be compared with the data in Figure 40 which shows that there is a strong correlation between vehicle delay times and the number of vehicle stops when the AI and the FCFS methods are used but there is no correlation found when the TLC method is used as shown in Figure 41.

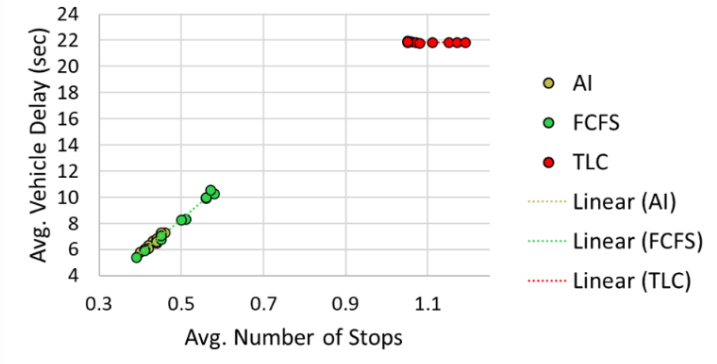


Figure 41 – Scatter graph that shows the correlation between vehicle delay times and the number of vehicle stops at an intersection based on the results obtained from mid traffic demand scenarios with major/major demand ratio.

8.2.3. Vehicle Speed

A speed limit of 50 km/h was applied in all validation scenarios for all vehicles. This meant that free-flow speed of all vehicles in traffic was expected to be near this speed limit when there was no congestion. Figure 42 shows the average vehicle speed for each traffic control method under all traffic demand scenarios. The AI and the FCFS methods gave similar results with a vehicle speed range of 42.43-46.18 km/h and 45.27-47.62 km/h under all mid and low traffic demand scenarios including both demand ratios. When the TLC method was used in the aforementioned scenarios, the average vehicle speed reached up to 40.66 km/h.

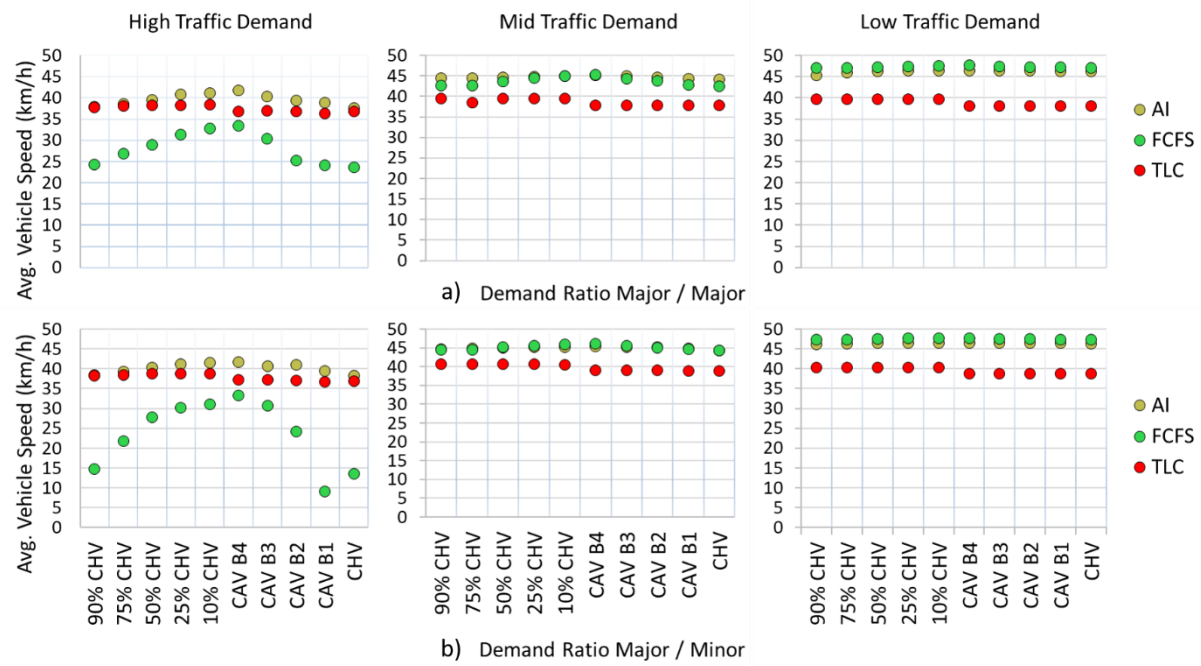


Figure 42 – Average vehicle speed for each driving behaviour and CAV penetration ratio under high, medium and low traffic demand scenarios a) when the demand ratio is major / major (top 3 graphs) and b) when the demand ratio is major / minor (bottom 3 graphs).

The data in Figure 42 also shows that free-flow vehicle speed breakdown occurs for scenarios with high traffic demand. In particular, average vehicle speed fluctuations were more pronounced when the FCFS method was used, and higher average vehicle speed was observed as CAV penetration rate increased or driving behaviour became more assertive.

It is also important to highlight that the AI and the TLC methods could successfully deal with two different demand ratios based on the fact that average vehicle speed decrease was no more than 4% when traffic demand was high, and the demand ratio was changed from major/major to major/minor.

However, the FCFS method was significantly worse in major/minor traffic demand ratio scenarios when the traffic demand was high with speed values dropping down to 9.06 km/h.

8.2.4. Queue Length

Average queue length of all approaching links for each control method is shown in Figure 43. It can be seen from this figure that it is common in all validation scenarios for the queue length to decrease as the CAV penetration rate increases or driving behaviour becomes more assertive. This trend was more pronounced as traffic demand increased. This outcome can be attributed to the standstill distance (CC0) parameter of driving behaviour model within Vissim. The expectation was that CAVs would stop with reduced gaps from each other in a queue as driving behaviour became more assertive, which in turn would reduce the queue length.

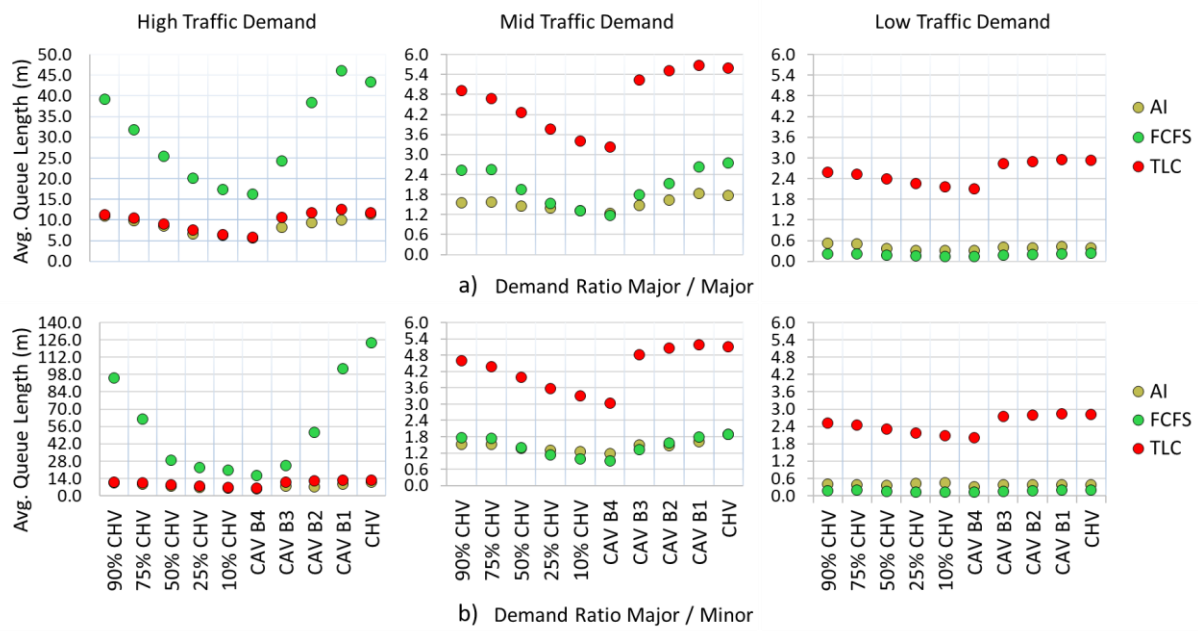


Figure 43 – Average queue length for each driving behaviour and CAV penetration ratio under high, medium and low traffic demand scenarios a) when the demand ratio is major / major (top 3 graphs) and b) when the demand ratio is major / minor (bottom 3 graphs).

When the traffic demand was low or mid, the TLC method had the greatest queue lengths under all scenarios with a maximum value of 5.68 m for CAV B1 driving behaviour. The impact of driving behaviour on queue length was more visible for the TLC method under low and mid traffic demand scenarios whereas the FCFS method showed the greatest variance in queue length values under high traffic demand scenarios. The AI method in general demonstrated the best performance in terms of

having the smallest queue length values for all except for the low traffic demand scenarios in which the FCFS method slightly outperformed the AI method.

8.2.5. Fuel Consumption

Average fuel consumption of all vehicles for the duration of each validation scenario is shown in Figure 44. The effects of CAV penetration rate or driving behaviour on fuel consumption became marginal as traffic demand decreased from high to low. The AI method consistently gave the lowest fuel consumption in all validation scenarios apart from the low traffic demand scenarios where the FCFS method was slightly better with a range of 0.86-1.62 gallons of fuel consumption difference between the two methods.

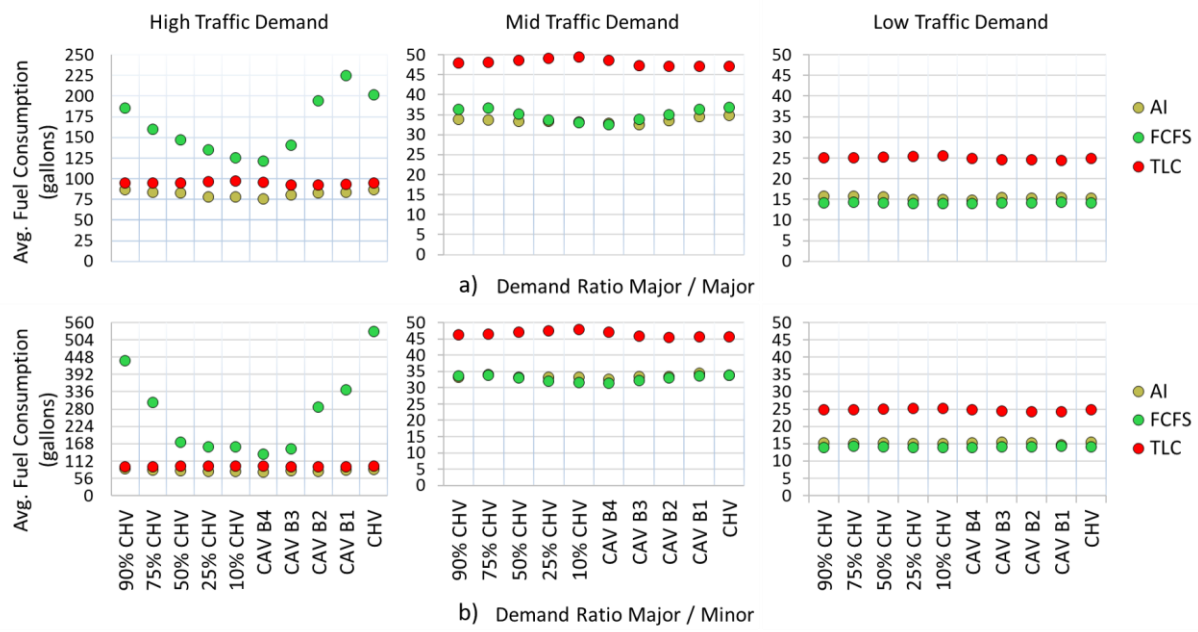


Figure 44 – Average fuel consumption for each driving behaviour and CAV penetration ratio under high, medium and low traffic demand scenarios a) when the demand ratio is major / major (top 3 graphs) and b) when the demand ratio is major / minor (bottom 3 graphs).

A comparison of the data in Figure 44 with Figure 40 shows that there is a strong correlation between the number of vehicle stops and fuel consumption for all validation scenarios. This relationship between the results is shown in Figure 45 for the mid traffic demand scenarios.

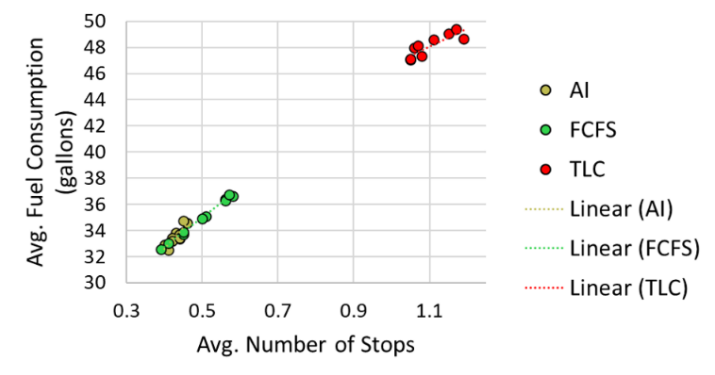


Figure 45 – Scatter graph that shows the correlation between fuel consumption and the number of vehicle stops at intersection based on the results obtained from mid traffic demand scenarios with major/major demand ratio.

8.2.6. Gas Emissions

Gas emission results collected during the simulation work included CO, NOx and VOC. Figure 46 shows the average CO emissions of all vehicles for the duration of each validation scenario. The same trend was observed in NOx and VOC results, and therefore, only CO emission results are presented in this section. The graphs for the rest of the gas emission results can be found in Appendix C.

As shown in Figure 46, CO emissions decreased for all control methods as traffic demand decreased from high to low. Similar to previous results, CAV penetration rate and driving behaviour had significant impacts on CO emissions in high traffic demand scenarios and this impact became marginal as traffic demand decreased.

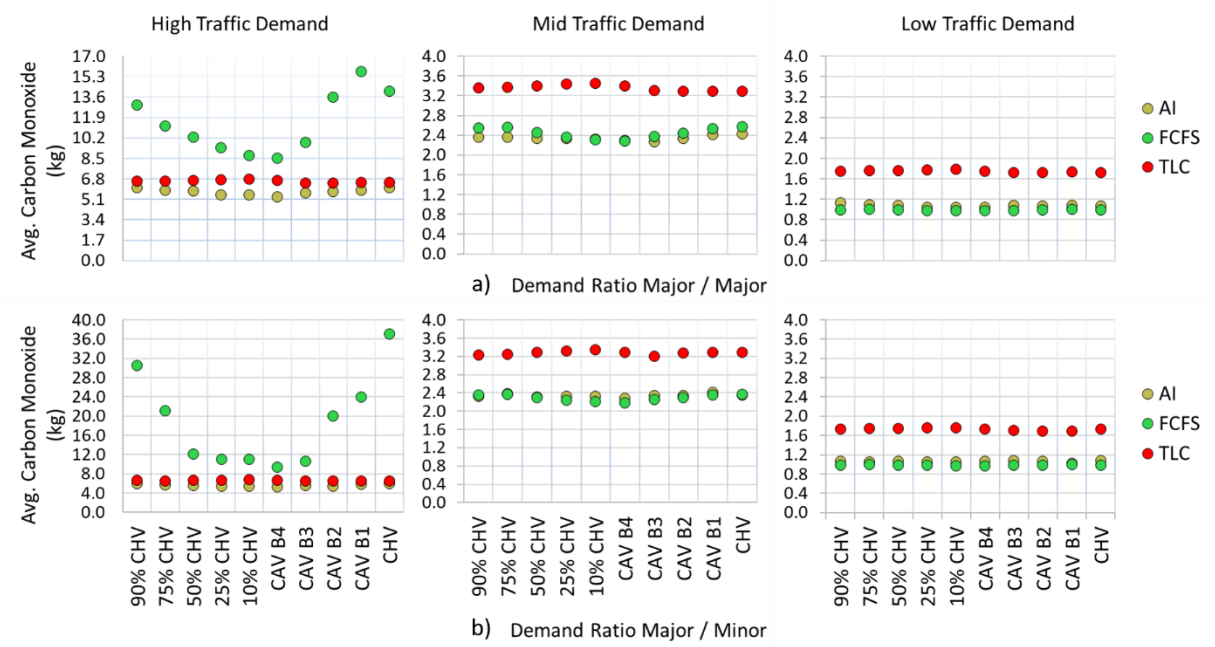


Figure 46 – Average carbon monoxide emission for each driving behaviour and CAV penetration ratio under high, medium and low traffic demand scenarios a) when the demand ratio is major / major (top 3 graphs) and b) when the demand ratio is major / minor (bottom 3 graphs).

A comparison of the data in Figure 46 with Figure 40 shows that there is a strong correlation between the number of vehicle stops and CO emissions for all validation scenarios. This relationship between the results is shown in Figure 47 for the mid traffic demand scenarios.

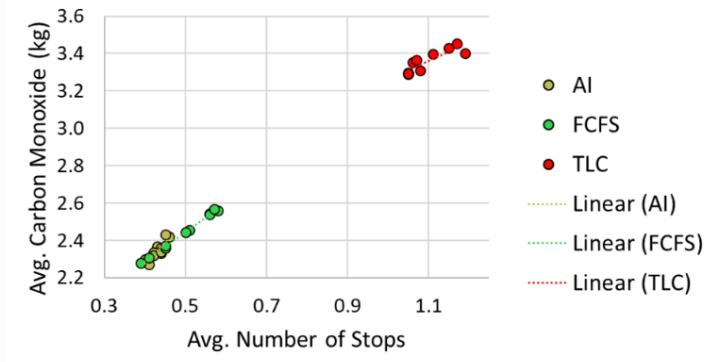


Figure 47 – Scatter graph that shows the correlation between CO emissions and the number of vehicle stops at intersection based on the results obtained from mid traffic demand scenarios with major/major demand ratio.

8.3. Scaled Testbed Experiment Results

In this section, the results of the scaled testbed experiments and its digital twin are presented. A comparison of the data obtained from the two platforms was expected to give similar results in terms of intersection throughput. On the other hand, a direct comparison with absolute performance metric values between the results of the simulation work (Section 8.2) and the scaled testbed experiments (Section 8.3) would not offer a meaningful data analysis. This is mainly due to the difference in the number of vehicles used for the simulation work (in the range of thousands) and the scaled testbed (in the range of tens). Instead, the experiment results are presented in terms of relative vehicle delay improvements in percentage between the AI and the TLC methods which can be considered as a normalisation technique to enable direct comparison of the experiment results.

8.3.1. Intersection Throughput

The experiments in the scaled testbed and the digital twin were run for 15 minutes for each scenario, and each scenario was repeated twice. The results represent the average values of all runs which is also scaled from 15 min to 1 hour by simply multiplying the obtained experiment result by 4. The intersection throughput data in Figure 48 makes a comparison between the scaled testbed and the digital twin results for TLC and the AI traffic control methods. The y-axis shows the intersection throughput in terms of number of vehicles and the x-axis has a series of discrete points that represent each validation scenario (See Chapter 7.2.5). The naming convention for the validation scenarios is *wCars_xS_yL_zR* where w denotes the traffic demand in total number of scaled cars and x, y, z denote the traffic demand at figure-of-eight intersection to go straight, turn left and turn right respectively. Furthermore, the percentage values on top of the data series Figure 48 indicate how far the scaled testbed results are from the digital twin results.

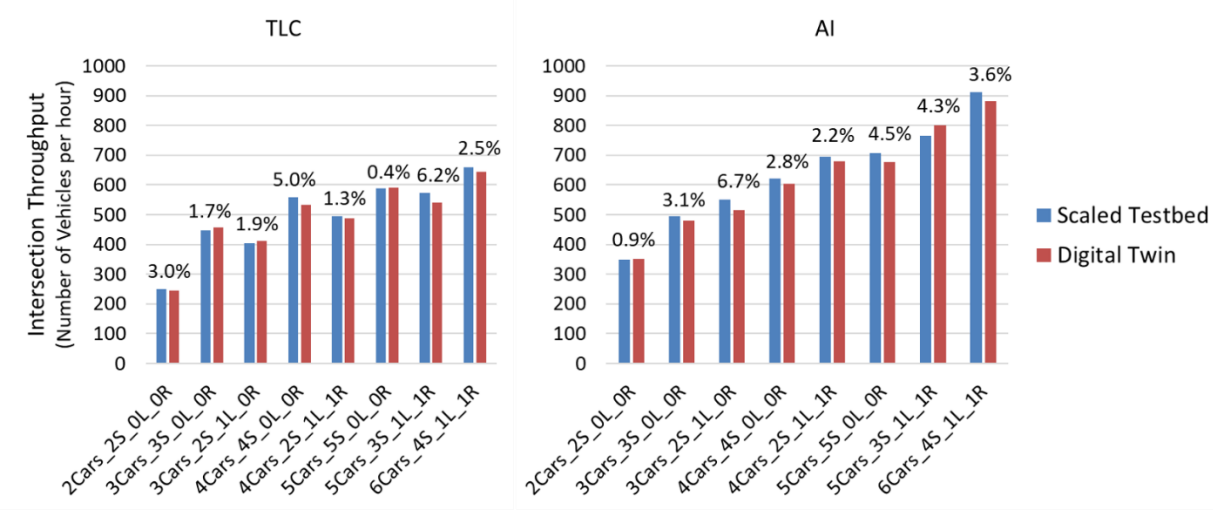


Figure 48 – Intersection throughput data that is obtained from the scaled testbed and the digital twin experiments for all scenarios under the AI and the TLC traffic control methods. The percentage values represent how far the scaled testbed results are from the digital twin results.

The most interesting aspect of the data in Figure 48 is the small variation between the scaled testbed and the digital twin results. The maximum intersection throughput difference was measured to be 6.2% for the TLC method and 6.7% for the AI method. This result is encouraging, and the evidence support that the driving behaviour of the scaled cars were implemented as close to CAV B1 as possible in the simulation environment. It is also shown in this figure that the intersection CrA was utilised more efficiently with the AI method under all traffic demand scenarios as the intersection throughput was consistently more that the TLC method and it reached a maximum of 912 veh/h, while it stayed at 660 veh/h with the TLC method.

8.3.2. Vehicle Delay

The average vehicle delay for the validation scenarios under the AI and the TLC methods in the scaled testbed is shown in Figure 49. CAV B1 driving behaviour was implemented in the scaled cars as explained in Section 7.2.4. It can be seen that the vehicle delay times increased as traffic demand increased for both control methods as expected. The data shows that the AI method consistently outperformed the TLC method in all scenarios in terms of vehicle delay times. The range of vehicle delay values was measured to be 69.1-400 sec with the AI method whereas this was measured to be 353.6-510 sec with the TLC method.

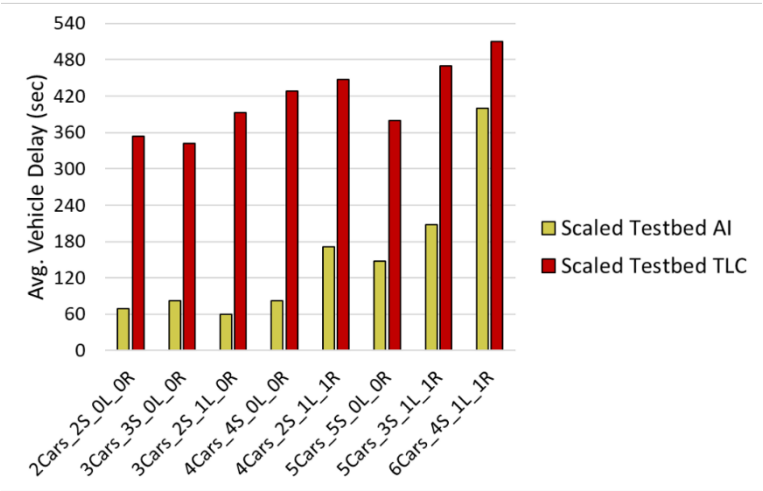


Figure 49 – Average vehicle delay data that was obtained from the scaled testbed experiments for all scenarios under the AI and the TLC traffic control methods.

The average vehicle delay percentage improvement when the AI method is used instead of the TLC method is calculated as: $\frac{t\_delay_{tlc} - t\_delay_{ai}}{t\_delay_{tlc}} * 100$ where t_delay_{tlc} and t_delay_{ai} denote the average vehicle delay data obtained when the TLC and the AI methods are used respectively. Based on this calculation, the vehicle delay improvement can also be calculated for the data obtained from the simulation work (See Figure 36). The rationale is to make a relative improvement comparison between the results obtained independently from the simulation work and the scaled testbed experiments.

Figure 50.d gives a summary of vehicle delay percentage improvements for each driving behaviour and CAV penetration ratio in the simulation work and the scaled testbed. Figure 50.a, Figure 50.b and Figure 50.c show, as an example, in three steps how the percentage improvement value for 90% CHV in Figure 50.d have been obtained. First of all, the average vehicle delays for the AI and the TLC methods are shown in Figure 50.a under high, mid and low traffic demand scenarios. The percentage value on top of the bars in Figure 50.a represent the vehicle delay improvement when the AI method is used compared to the TLC method. Following this, in Figure 50.b, the percentage improvement values are displayed as a scatter graph to highlight the range of percentage improvement. Finally, Figure 50.c displays the improvement range as a box plot where median and mean values are marked. The other percentage improvement values in Figure 50.d have been calculated in the same way.

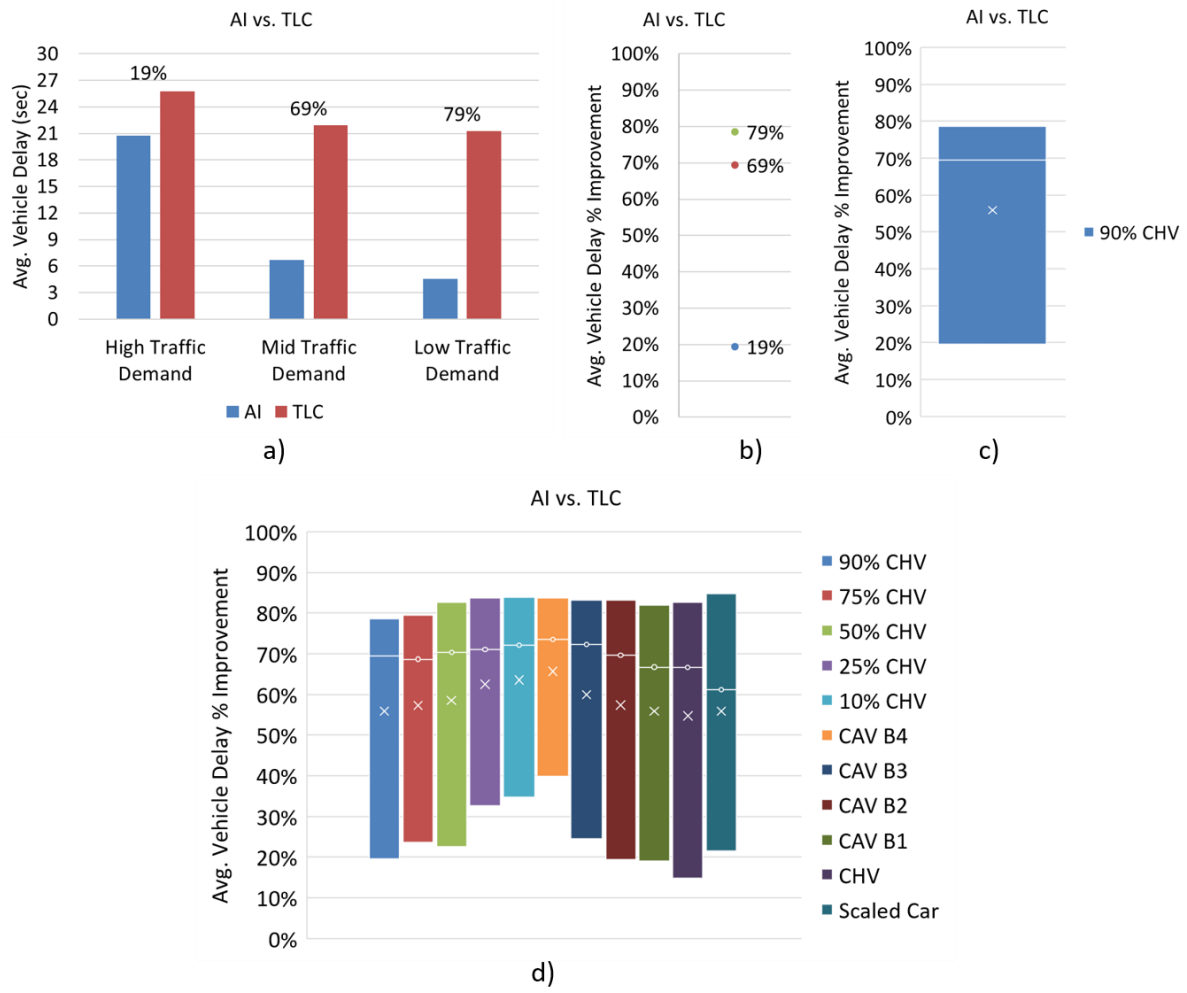


Figure 50 – a) Average vehicle delay comparison between the AI and TLC methods for 90% CHV scenarios. b) Average vehicle delay percentage improvement is shown in a scatter graph to highlight the improvement range. c) The scatter graph in b) is transformed into a box plot. The middle line of the boxes and the x inside the boxes represent the median and mean values respectively. d) Average vehicle delay percentage improvement is shown for each driving behaviour in simulation work and the scaled testbed when the AI method is used compared to the TLC method. The range of percentage values represent the results obtained from all traffic demand scenarios.

The scaled car percentage improvement range in all 8 validation scenarios is shown as 22-85% in Figure 50.d which is labelled as Scaled Car. As explained previously, the scaled cars were essentially calibrated to have similar driving behaviour as CAV B1, and a closer inspection of Figure 50.d shows that the range of vehicle delay improvements obtained in Vissim for CAV B1 is 19-82% which has a strong correlation with the results of the scaled testbed. The vehicle delay percentage improvement became better as CAV penetration rate increased or as the driving behaviour became more assertive.

8.4. Discussion and Key Findings

The validation scenarios were constructed to measure and analyse the impacts of traffic control methods on traffic flow, congestion, journey times and environment. The results obtained from the simulation work and the scaled testbed combined together provided extremely useful insights on the performance comparison of different traffic control methods in realistic real-world scenarios. Based on the comprehensive results presented in this chapter, particular key findings are explained below.

Higher CAV penetration rate brings various benefits as long as more advanced traffic control methods are used

The results of this research work indicate that the penetration of CAVs into the traffic flow will bring variety of benefits in terms of reduced vehicle delays, congestion, fuel consumption and gas emissions. These benefits were shown to increase as the CAV penetration ratio increased. Another important finding was that the aforementioned benefits highly depended on the traffic control method used. When the TLC method was used, no significant benefits (<10% under all scenarios) were seen in terms of vehicle delays whereas the AI method offered 7.61% reduced vehicle delays even when there was only 10% CAV penetration rate. It can thus be suggested that unless more advanced traffic control methods are used, such as the proposed AI method in this work, no significant benefits can be gained with increased CAV penetration rates which is also supported by Atkins (2016b) and Fagnant and Kockelman (2015).

Driving behaviour choice can have significant impact on traffic flow

The literature review highlighted the fact that CAVs with different driving style configurations will exist in traffic as CAV penetration rate increases (Atkins, 2016c). The range of driving behaviour is not necessarily expected to be more assertive than a human driver, and therefore, some CAVs may represent driving behaviour more cautious than a human driver. The evidence found in this work suggests that marginal benefits can be obtained in terms of vehicle delay times under mid or low traffic flow conditions when the traffic flow is 100% CAVs that are more cautious than a human driver, and more importantly, the delay times can be worse than a human driver when the traffic is congested under high traffic demand. Therefore, the driving behaviour should be a key consideration when rolling out CAVs into our road networks as the driving style will have an influence on the relative benefits compared to CHVs in terms of vehicle delays and congestion among many other factors that will determine the roll out of CAVs.

Marginal benefits in low traffic demand

The results presented in this chapter show that benefits in terms of vehicle delay, congestion, preventing stop-and-go movements, fuel consumption and gas emissions were much greater when traffic demand is high. Under low traffic demand conditions, all traffic control methods showed marginal improvements in all performance metrics as CAV penetration rate increased or CAV driving behaviour became more assertive. This finding has important implications when deciding the initial mass-scale deployment strategy for any automated public transportation services i.e. buses, taxies etc. In other words, the evidence suggests that urban areas and congested networks will benefit more from having CAVs in the traffic flow.

There is no single traffic control method that fits all real-world scenarios

The AI traffic control method has shown to perform the best, based on all performance metrics, in all traffic conditions apart from the low traffic demand scenarios where the FCFS method performed on a par with or slightly better than the AI method. Considering that there are development, installation and maintenance costs associated with each control method, the findings in this study suggest that even simple heuristic-based control methods, such as FCFS, can be implemented as a traffic control method in areas that do not experience high volumes of traffic. This could be attractive to local authorities and transport service providers in terms of keeping the cost and complexity of the control system to minimum.

An AI-based traffic control can adapt to changes in traffic flow

This research work demonstrated that an AI-based control method can perform well in a traffic control domain which is stochastic in nature, and it can adapt to unseen states and conditions during training. It is therefore likely that once an AI neural network is trained under a simulation environment, it can be deployed to multiple locations without requiring a special training procedure for every single intersection. It is also important to bear in mind that 4-way junction and 4-way roundabout geometries were used in this work during validation. Therefore, there is limited evidence to suggest that the AI control method could work on geometries with a different number of lanes and links when it is trained on 4-way junction type only. In theory, it is possible to train an AI network to operate in multiple intersection geometries as long as the training scenarios are setup appropriately for this objective.

A scaled testbed coupled with a digital twin in simulation can accelerate the development and the validation of advanced traffic control methods

Simulation tools can offer great benefits in terms of avoiding time-consuming and expensive experiments with physical assets. On the other hand, practical testing and validation is also not avoidable mainly due to the fact that a thorough understanding of a traffic environment for modelling in simulation including all factors involved is very challenging, if not impossible. This research work demonstrated that it was possible to set up a cost-effective scaled testbed with scaled cars in order to generate traffic flow with CAVs and validate the operations of advanced traffic control methods. The digital twin approach was extremely useful when making associations with the simulation experiments in terms of results obtained. The present work suggests the possibility that much larger scaled testbeds where multiple intersections and road types are implemented can help towards understanding the impacts of CAVs in traffic flow at a fraction of the cost and duration that would otherwise be required in a real world setting.

8.5. Summary

The validation results from the simulation work and the scaled testbed experiments are presented in this chapter. The impacts of different traffic control methods on traffic flow, congestion, journey times and environment are quantified with performance metrics. Key findings are drawn out from the comprehensive experiment results and a discussion on these findings is provided. Together these findings provide important insights into enabling CAV support from an infrastructure point of view. The next chapter moves on to presenting the conclusions of this work together with recommended future research directions in order to expand the ideas and methods proposed in this work.

Chapter 9

9. Conclusions and Future Work

9.1. Research Summary

In this thesis, a novel traffic control method has been proposed that integrates cooperative ITS features and supports the integration of CAVs into mixed-fleet operations to make better use of the road transport infrastructure. To this end, an extensive literature review has been presented in Chapter 2 in traffic control methods, focusing on the recent advancements in CAVs, AI techniques and cooperation between traffic users via wireless communications. The traffic control problem definition, the V2I communication protocol, the RL algorithm set up and training, and the validation methodology has provided a series of insights into the future use of the road networks, vehicles and transport services that will allow the delivery of a positive disruptive change on how traffic is managed at intersections. The main conclusions derived from this research work are summarised below:

The AI traffic control method with C-ITS communication features achieves less congestion, journey time, fuel consumption and gas emissions when compared to the fixed-time signalised control method and the heuristic rule-based FCFS method.

Chapter 3 has presented the unsignalised traffic control problem statement and the V2I wireless system details including the communication protocol and the data requirements under certain assumptions in order to focus on the main contributions of this research work. The proposed unsignalised traffic control is one of the ITS applications that targets the reduction of congestion whilst preventing collisions. Therefore, the communication latency between the traffic users was considered to be less than 100ms as specified in SAE J2735 standard. The communication protocol presented in Chapter 3 forms the foundation for the proposed unsignalised intersection control, enabling bi-directional data exchange between the vehicles and the infrastructure.

The ICA, in the context of RL has been presented in Chapter 4. The main task of this agent has been to determine the individual vehicle priorities based on the objective of reducing average vehicle delays on all approaching links of the intersection. To this end, the information received from the approaching

vehicles by using the communication protocol from Chapter 3 has been instrumental. The current traffic observation vector has been constructed with this information to capture the dynamics of the traffic state. This input vector is essentially the "eyes" and the "ears" for the traffic control agent to make sense of the current traffic state and take actions accordingly. Chapter 4 has also established the conflict resolution method in order to calculate the crossing time windows for all vehicles waiting to cross the intersection. The conflict resolution method takes the vehicle priority list, generated by the agent, as an input and produces safe crossing time windows so that collision avoidance is ensured during intersection crossing.

The results obtained from the simulation work in Chapter 6 and the scaled testbed in Chapter 7 combined together have provided extremely useful insights on the performance comparison of different traffic control methods in realistic real-world scenarios. The impacts of different traffic control methods on traffic flow, congestion, journey times and environment have been quantified with performance metric, and key findings have been drawn out from the comprehensive experiment results in Chapter 8. Together these findings have provided important insights into enabling CAV support from infrastructure point of view. The key findings can be summarised as below and suggest that the proposed AI traffic control method outperforms the benchmarked control methods in all identified performance metrics:

- An AI-based traffic control method can adapt to changes in traffic flow and generalise its strategy well to unseen traffic states.
- Higher CAV penetration brings various benefits as long as more advanced traffic control methods are used.
- Driving behaviour choice can have significant impact on traffic flow and congestion.
- Marginal benefits have been gained from of having more assertive driving behaviour or more CAV penetration rate in low traffic demand with all traffic control methods.
- There is no single traffic control method that fits all real-world scenarios.
- A scaled testbed coupled with a digital twin in simulation can accelerate the development and the validation of an advanced traffic control method.

The stochastic nature of traffic environment has significant implications on how a traffic control algorithm, based on RL techniques, should be designed and trained.

The details of the AI algorithm and methods have been presented in Chapter 4. Traffic is stochastic in nature, meaning the reaction of the environment might not be predicted precisely. In addition, traffic environment is one of the prime examples where the traffic control actions affect the flow of vehicles

gradually in time rather than immediately after taken a particular action. This has led us to explore and implement methods that can handle stochasticity and delayed outcome.

The state representation, action space and the reward mechanism have all been combined together under the TD3 algorithm. The agent training methodology and the associated parameters and configurations have been explained in Chapter 5. The main objective of the training procedure has been to enable the agent to experience all potentially possible situations in the traffic environment so that it could learn what action sequences result in better policy. Model-free RL algorithms like TD3 are sample-inefficient, meaning they require a lot of interactions to learn a good policy. The exploration and exploitation dilemma section in Chapter 5 explained the techniques implemented in order to overcome the sample-inefficiency challenge and reduce the training time. Filling the experience replay buffer initially with the traffic light policy has been found to be useful in terms of faster convergence to the optimal policy.

In order to validate and test advanced traffic control methods in real-world traffic scenarios with hundreds or thousands of CAVs, in a cost-effective way, in a controlled and repeatable environment, it is essential to consider the integration of virtual elements and physical assets.

The penetration rate of CAVs is too small, worldwide as of today, to gather any real-world evidence about their impact on traffic flow. Therefore, any research work in this field of research utilises simulation tools to model their behaviour. In this work, this approach has also been taken where various different driving styles have been modelled in conjunction with traditional human driving behaviour in order to simulate mixed-driving scenarios. The traffic simulation methodology and the tool chain have been presented in Chapter 6 and the scaled testbed setup has been presented in Chapter 7. The selected tools are state-of-the-art solutions which are widely used within the machine learning and traffic engineering fields both in academia and industry.

The scaled testbed introduced in Chapter 7 is a scaled road network in a figure of eight shape with a single road lane. It has been constructed with an objective of replicating real-world traffic control scenarios in a scaled, cost-effective and controlled environment. It can be seen as a bridge between a simulation work and a real-world deployment of such a system. Realistic environmental cues have been included in the scaled testbed such as V2I communications, road markings and intersection crossing shared space so that the impact of an unsignalised traffic control can be obtained and compared with a traffic light based control method. Furthermore, the digital twin of the scaled testbed has been also created in Vissim to cross validate the experiment results.

The technical details of the scaled CAVs have also been presented in Chapter 7. The NVIDIA Jetson Nano processor has been used as the main computing platform which runs the software for the automated driving application. The automated driving task required a training data collection process with a human driver controlling one car remotely around the testbed. 10 cars have been assembled that were built to the same specification, and they could drive autonomously around the scaled testbed simultaneously.

9.2. Original Contributions

This research work advances the current state-of-the-art in traffic control at intersection further by making the following main contributions:

C1. Proposal of a novel centralised and unsignalised traffic control method based on the Twin-Delayed Deep Deterministic Policy Gradient (TD3) RL algorithm for mixed-fleet operations where CAVs and Connected Human-driven Vehicles (CHV) co-exist in traffic. The proposed method achieved up to 84% less average vehicle delays and 41% less fuel consumption during intersection crossing compared to the fixed-time TLC method both in simulation and practical experiments.

C2. Proposal of a novel state representation and a reward mechanism for the traffic control agent which are put together under the TD3 algorithm as part of the RL framework. The state space has 8 values for each road lane approaching an intersection that consist of the average traffic flow parameters and the lead vehicle parameters. The proposed state space captures the traffic environment state as comprehensive as possible while keeping the number of variables in the vector to a minimum. The reward mechanisms has been structured in a way to reduce average vehicle delays at the intersection whilst considering safety and platoon formations.

C3. Creation of a training and validation software platform for an AI-based traffic control method. The software platform consists of the state-of-the-art Vissim traffic simulation tool, TensorFlow open source machine learning library and National Instruments (NI) LabVIEW tool. The platform brings together the aformentioned tools that complement each other in order to generate a realistic traffic environment and scenarios where CAV penetration rates, traffic demand levels, driving styles and road geometry are varied systematically in a repeatable way. The software platform has been

created by comprehensive coding work that includes three programming languages, C++ (Vissim), Python (TensorFlow) and Visual Programming (NI LabVIEW).

C4. Development of a scaled testbed with realistic cues about the traffic environment together with multiple scaled CAVs. The digital twin of the scaled testbed has also been created in the simulation environment for cross-validation of the scaled testbed experiment results. This validation and testing approach can be seen as an extremely useful intermediate step when taking a complex system such as traffic control from simulation to real-world deployment in a cost-effective and controlled way. The experiment results obtained from the scaled testbed and its digital twin indicated strong similarities, based on a maximum difference of 6.8% in intersection throughput metric between both platforms, which proved the relaibility of the scaled testbed approach with the scaled CAVs.

The original contributions of this research work are expected to influence the new ways of thinking about the use of road networks, vehicles and transport services that will allow delivering positive disruptive change on how traffic is managed at intersections whilst providing the catalyst for new experiments in the future of mobility.

9.3. Future Work

This section highlights some particular open research questions and the knowledge gap that remains, and finally makes recommendations for future research in this field. Unsignalised traffic control with reinforcement learning in the presence of CAVs represents an integration of emerging technology areas, and therefore, there is definitely scope for substantial future work. This is specifically relevant for intelligent transport planners, road network operators, local authorities and ITS technology providers where increased knowledge of the potential for such advanced control systems can allow for a more accurate assessment of their potential impacts on the traffic flow. The recommendations for future work are summarised below:

Mixed-fleet operations

CAVs will be operating in traffic with human drivers for a foreseeable future where the number of scenarios that can be encountered is practically infinite, as human drivers tend to find alternative ways to deviate from expected behaviour in traffic. For this reason, the mixed-fleet operations should be understood thoroughly when considering a disruptive technology like unsignalised intersection control. As of today, the work towards having formalised standards or approval schemes for the integration of

CAVs into the transport system is still in its infancy. In order for safe and rapid adoption of unsignalised traffic control method, mixed-fleet operations should be initially modelled in simulation environments for complex driving scenarios. These scenarios should include but not limited to the impacts of CHVs not following the instructions or traffic rules.

Multi-intersection operations in the context of multi-agent systems

In this work, a single intersection control has been studied without considering the wider impacts of upstream or downstream traffic intersections. The management of multi-intersection network with a single agent can be challenging, and therefore, consideration for multi-agent system should be made where agents synchronise their activities and make decisions jointly to meet the design objectives such as keeping local area congestion under a certain level. In a multi-intersection network, when traffic demand is high, congestion in one intersection can have a knock-on effect ono other neighbour intersections. Thus, an efficient control method that maximises a collaborative long-term reward between neighbour intersections is desired. The training methodology for such scenarios for neural networks is also an area that is recommended for future research where neighbour intersections share traffic state data with each other for more pro-active decision making.

Implications on field deployment and commercialisation

One of the key considerations of integrating unsignalised traffic control as a key transportation service is to determine what is the optimal positioning of such technology in the immediate and longer term future. In creating such a disruptive and novel traffic control method, the question of enabling infrastructure investment and deployment challenges rises. Rigorous testing procedure must be in place prior to deployment of such a system that considers CAV operations and any edge cases so that rare occurrence of a particular situation would not jeopardise the safety of the system and the traffic users. Further study in this area is definitely needed to guide the integration of an unsignalised traffic control system into the transportation network and how the associated government and industry business cases for commercialisation could be structured.

Consideration for other type of road users

In this work, certain assumptions have been made in order to focus on the isolated impacts of the proposed unsignalised traffic control method on the traffic flow. One of the assumptions that must be relaxed and considered in future work is the inclusion of vulnerable road user such as pedestrians, cyclists, and other types of vehicles such as buses, lorries etc. As CAV penetration rate increases in the

future, they are likely to operate in increasingly complex traffic environments. It is therefore essential to understand the interaction between CAVs and other road users that has not been studied in this work.

The impact of unsignalised traffic control on safety and user acceptance

Safety and user acceptance are two key drivers when it comes to scaling up a disruptive technology. The Vissim simulation tool that has been used in this study does not include a built-in safety assessment and analysis software component. Therefore, no safety impacts could have been measured in this research work. Further work in simulation is recommended that integrates safety assessment tools for collision risk analysis. In parallel to this, it will also be necessary to gain user trust, and this can be achieved by proving safety of the traffic control system in all potential real-world scenarios with methodical test and validation procedures.

Closed-loop operation of priority assignment and conflict resolution stages of the control algorithm

Conflict resolution stage of the proposed traffic control algorithm in this research work receives a list of assigned priorities by the TCA, and following this, crossing time windows are allocated for each vehicle. It might be possible in the future to feedback the output of conflict resolution stage back to the reward mechanism for the TCA to develop a full picture of the consequences of its actions.

Cybersecurity

Further reseach into solutions for secure connectivity between RSUs and CAVs/CHVs on the road is recommended as it has been assumed in this reseacrh work that the V2I connectivity and data was secure. The function of the RSU is to broadcast the traffic control, map and safety related data to all vehicles at the traffic intersection. Potential security threats to this communication link include spoofing the RSU communication unit to send false information to the approaching vehicles and signal jamming. The techniques that could be used to mitigate against these threats should be investigated for unsignalised traffic control.

Appendices

Appendix A

A1. Signal Time Optimisation Process Steps in Vissim (from the User Manual)

Vissim repeatedly runs simulations of the entire network during the optimisation process. The optimisation process is continued as long as changes in green times of the stages lead to an increase in the traffic flow or to a reduction in the average vehicle delay. The stage lengths with the best result have the highest flow and the lowest average vehicle delay and are stored in a file after the optimisation. The process steps are given below:

1. Vissim determines the average delay of all vehicles that have passed through the intersection, automatically evaluating each signal group over the entire simulation run.
2. For optimisation, the signal group in which the vehicles have the highest delay is determined for each stage.
3. The stage with the lowest maximum average delay is selected as the best stage.
4. The stage with the highest maximum average delay is selected as the worst stage.
5. A second of green time is deducted from the best stage.
6. A second of green time is added to the worst stage.
7. If a second can no longer be deducted from the best stage, the second best stage is used. If this can no longer be shortened, the next worst stage is always taken iteratively. If no other stage can be shortened, the optimisation is terminated.
8. A signal program is considered to be better than another if one of the following criteria is met:
    - If the flow formed by the total number of vehicles driven through the node during the simulation run has increased significantly by at least 25 vehicles or by 10% if this is less.
    - If the flow has not significantly decreased by 25 vehicles or by 10% and the average delay across all vehicles has decreased.
9. If a signal program is better than the best rated, it replaces this as the best. The optimization is continued with the next step.
10. The optimisation is terminated if one of the following criteria is met:
    - Once the signal program does not improve within 10 simulation runs.
    - Once the flow decreases by more than 25% compared to the best signal program.
    - Once the average delay increases by more than 25%.

Appendix B

B1. BOM for A Single Scaled Car

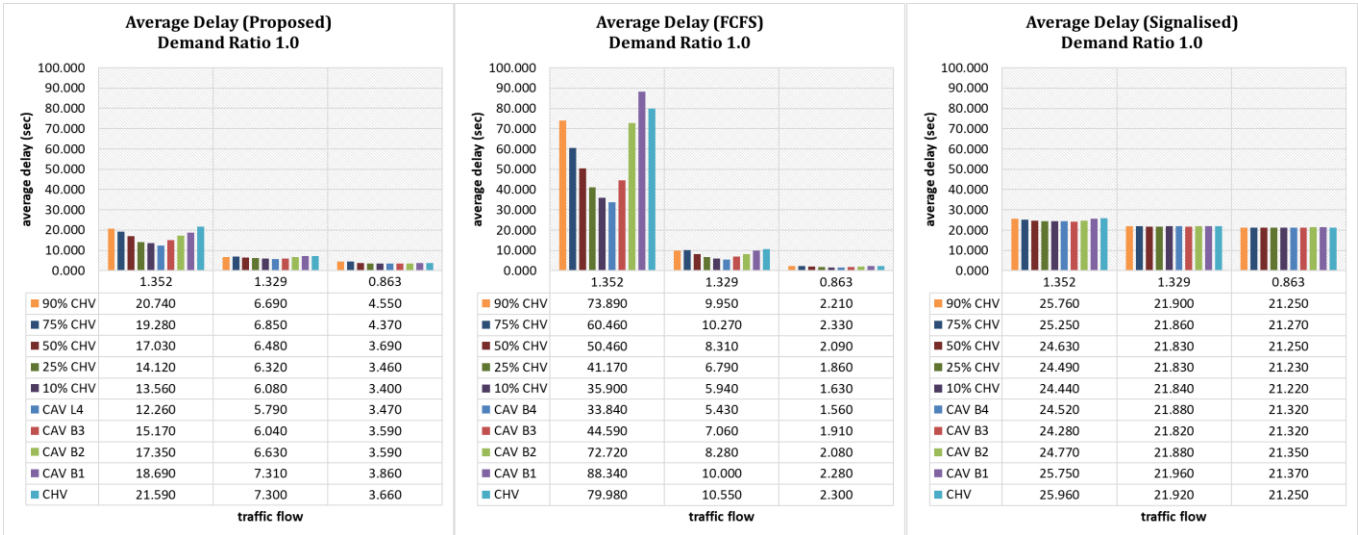
Part #	Part Name	Description	Qty	Supplier	Unit Cost	Cost
945-13450-0000-100	NVIDIA Jetson Nano Dev Kit	NVIDIA JETSON NANO 4GB DEVELOPER KIT (B01)	1	Silicon Highway	£ 82.50	**£ 82.50**
SSCF-M2-6-A2-R360	M2 X 6mm Full Thread Cap Head Screws (DIN 912) - Thread Locking A2 Stainless Steel	M2 Metric Thread Size (2mm), Length (L): 6mm, Fully Threaded: Yes	4	Accu	£ 0.22	**£ 0.88**
SSCF-M2.5-12-A2	M2.5 X 12mm Full Thread Cap Head Screws (DIN 912) - Thread Locking A2 Stainless Steel	M2.5 Metric Thread Size (2.5mm), Length (L): 12mm, Fully Threaded: Yes	8	Accu	£ 0.28	**£ 2.24**
HPN-M2.5-A2	M2.5 Hexagon Nuts (DIN 934) - A2 Stainless Steel	A2-70 cold-worked Stainless Steel, also known as 18-8 or 304 Stainless Steel	8	Accu	£ 0.03	**£ 0.24**
HPW-M2.5-A2	M2.5 Form A Flat Washers (DIN 125) - A2 Stainless Steel	A2 Stainless Steel, also known as 18-8 or 304 Stainless Steel	8	Accu	£ 0.03	**£ 0.24**
SDSQXAF-032G-GN6MA	SanDisk 32GB Extreme A1 V30 Micro SD Card (SDHC) + Adapter - 90MB/s	microSDHC™ (32GB), microSDXC™ (64GB-256GB)	1	MyMemory	£ 13.99	**£ 13.99**
N/A	Logitech C525 Portable HD Webcam	Logitech C525 Portable HD Webcam with Fast Autofocus	1	Amazon	£ 54.99	**£ 54.99**
136-3088	Edimax Bluetooth, WiFi USB 2.0 Dongle	Edimax 2-in-1 N150 Wi-Fi and Bluetooth 4.0 Nano USB Adapter	1	RSOnline	£ 16.96	**£ 16.96**
N/A	Adafruit 16-Channel 12-bit PWM/Servo Driver - I2C interface - PCA9685	12-bit resolution for each output - for servos, 4us resolution at 60Hz update rate	1	The PiHut	£ 14.00	**£ 14.00**
RB-Spa-1388	HC-SR04 Ultrasonic Range Finder	Ultrasonic sensor distance measuring module	2	RobotShop	£ 3.65	**£ 7.30**
N/A	Adafruit Premium Female/Female Jumper Wires - 20 x 3" (75mm)	N/A	1	The PiHut	£ 2.00	**£ 2.00**
N/A	HOBBYMATE Lipo Battery Safe Bag	20 x 15 x 15 cm	1	Amazon	£ 5.41	**£ 5.41**
					Total	**£ 200.75**

B2. BOM for the Scaled Testbed

Part #	Part Name	Description	Qty	Supplier	Unit Cost	Cost
N/A	EVA Foam (96 square feet / 24 tiles)	EVA Foam Play Puzzle Mat Interlocking Exercise Tiles Floor Mat Grey	96	ebay	£ 0.40	**£ 38.29**
N/A	Economy Matt Gaffer Tape (White)	Matt (non-reflective finish), high strength adhesive	4	GafferTape	£ 6.20	**£ 24.80**
N/A	Fluorescent Economy Matt Gaffer Tape (50m)(Fluorescent Orange)	Anti-Reflection Coating, glows under artificial UV/Black light	1	GafferTape	£ 8.26	**£ 8.26**
N/A	NETGEAR Nighthawk Smart Wifi Router (R7000) - AC1900	NETGEAR Nighthawk Smart Wifi Router (R7000) - AC1900	1	Amzon	£ 103.98	**£ 103.98**
N/A	Samsung 43" TU7100 HDR Smart 4K TV with Tizen OS [Energy Class A]	Samsung 43" TU7100 HDR Smart 4K TV with Tizen OS [Energy Class A]	1	Amazon	£ 319.00	**£ 319.00**
N/A	Logitech C922 Pro Stream Webcam	Logitech C922 Pro Stream Webcam, Full HD 1080p	2	Amazon	£ 119.99	**£ 239.98**
					Total	**£ 734.31**

Appendix C

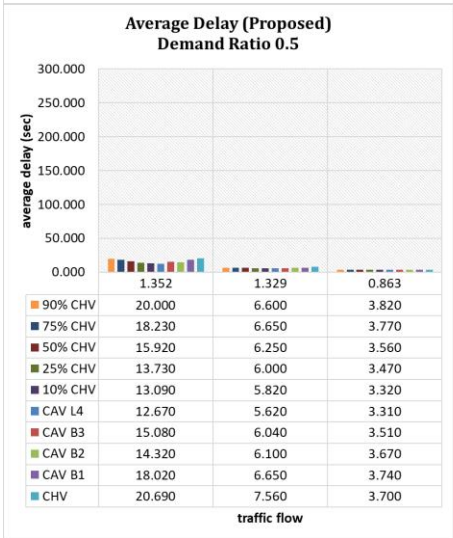
C1. Simulation Results on Average Vehicle Delay



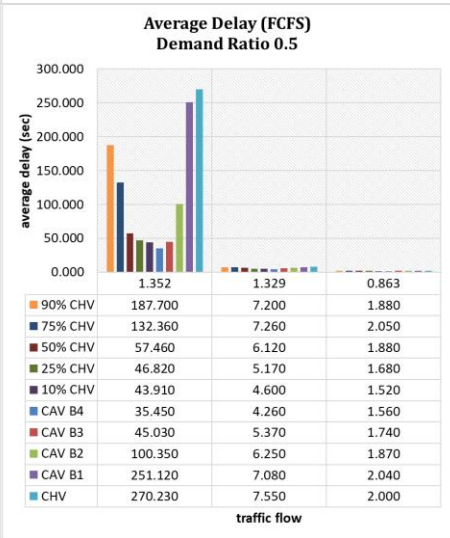
	1.352	1.329	0.863
90% CHV	20.740	6.690	4.550
75% CHV	19.280	6.850	4.370
50% CHV	17.030	6.480	3.690
25% CHV	14.120	6.320	3.460
10% CHV	13.560	6.080	3.400
CAV L4	12.260	5.790	3.470
CAV B3	15.170	6.040	3.590
CAV B2	17.350	6.630	3.590
CAV B1	18.690	7.310	3.860
CHV	21.590	7.300	3.660

	1.352	1.329	0.863
90% CHV	73.890	9.950	2.210
75% CHV	60.460	10.270	2.330
50% CHV	50.460	8.310	2.090
25% CHV	41.170	6.790	1.860
10% CHV	35.900	5.940	1.630
CAV B4	33.840	5.430	1.560
CAV B3	44.590	7.060	1.910
CAV B2	72.720	8.280	2.080
CAV B1	88.340	10.000	2.280
CHV	79.980	10.550	2.300

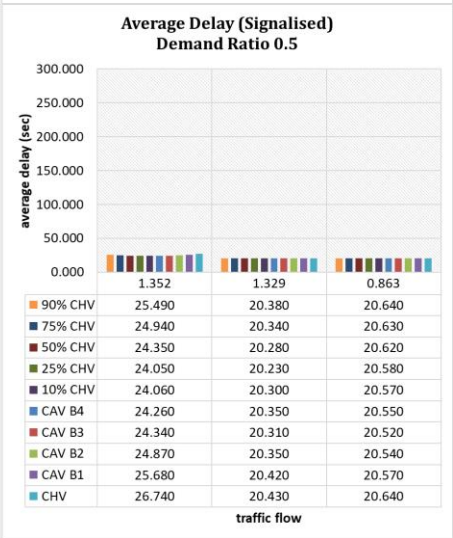
	1.352	1.329	0.863
90% CHV	25.760	21.900	21.250
75% CHV	25.250	21.860	21.270
50% CHV	24.630	21.830	21.250
25% CHV	24.490	21.830	21.230
10% CHV	24.440	21.840	21.220
CAV B4	24.520	21.880	21.320
CAV B3	24.280	21.820	21.320
CAV B2	24.770	21.880	21.350
CAV B1	25.750	21.960	21.370
CHV	25.960	21.920	21.250



	1.352	1.329	0.863
90% CHV	20.000	6.600	3.820
75% CHV	18.230	6.650	3.770
50% CHV	15.920	6.250	3.560
25% CHV	13.730	6.000	3.470
10% CHV	13.090	5.820	3.320
CAV L4	12.670	5.620	3.310
CAV B3	15.080	6.040	3.510
CAV B2	14.320	6.100	3.670
CAV B1	18.020	6.650	3.740
CHV	20.690	7.560	3.700

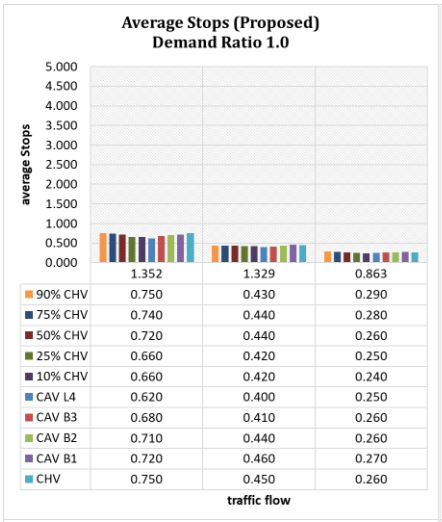


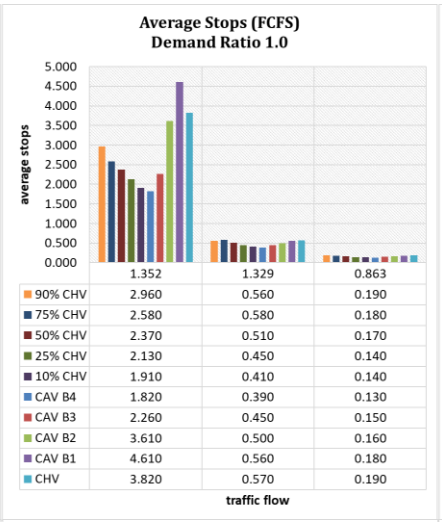
	1.352	1.329	0.863
90% CHV	187.700	7.200	1.880
75% CHV	132.360	7.260	2.050
50% CHV	57.460	6.120	1.880
25% CHV	46.820	5.170	1.680
10% CHV	43.910	4.600	1.520
CAV B4	35.450	4.260	1.560
CAV B3	45.030	5.370	1.740
CAV B2	100.350	6.250	1.870
CAV B1	251.120	7.080	2.040
CHV	270.230	7.550	2.000

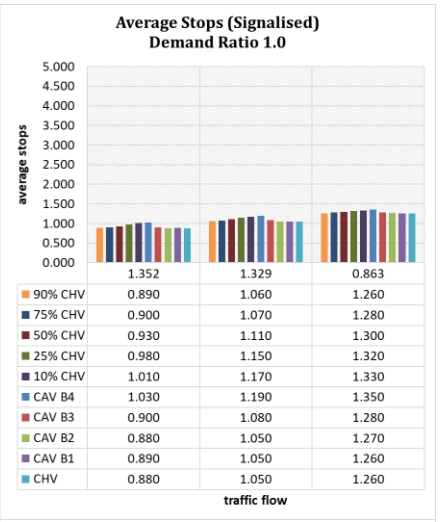


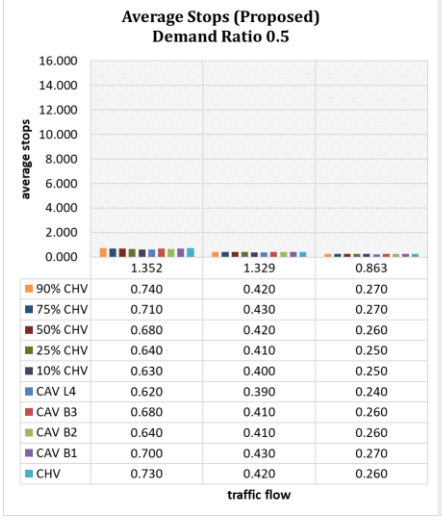
	1.352	1.329	0.863
90% CHV	25.490	20.380	20.640
75% CHV	24.940	20.340	20.630
50% CHV	24.350	20.280	20.620
25% CHV	24.050	20.230	20.580
10% CHV	24.060	20.300	20.570
CAV B4	24.260	20.350	20.550
CAV B3	24.340	20.310	20.520
CAV B2	24.870	20.350	20.540
CAV B1	25.680	20.420	20.570
CHV	26.740	20.430	20.640

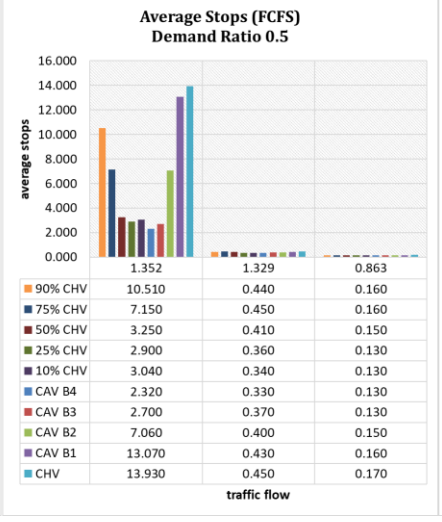
C2. Simulation Results on Average Number of Stops

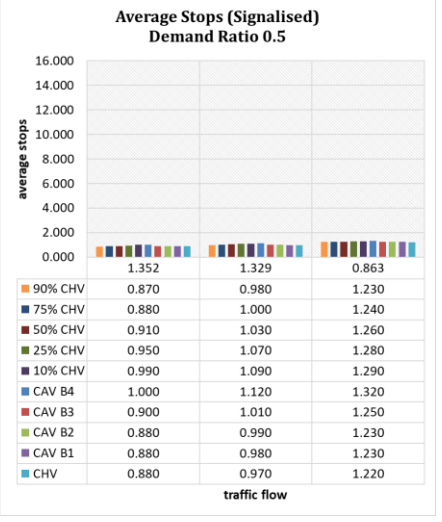




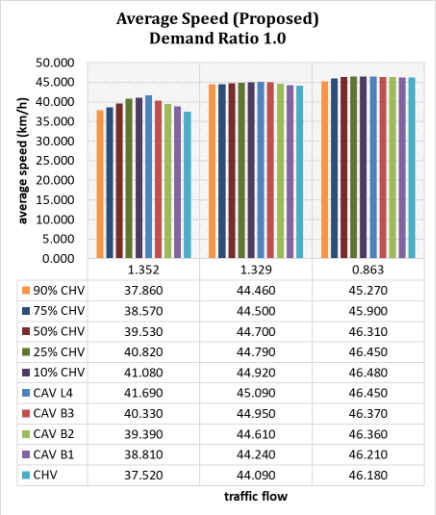




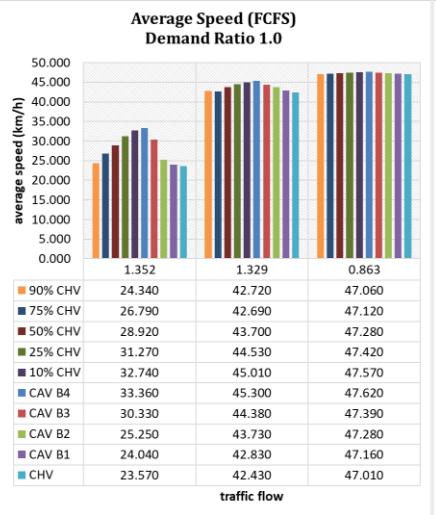




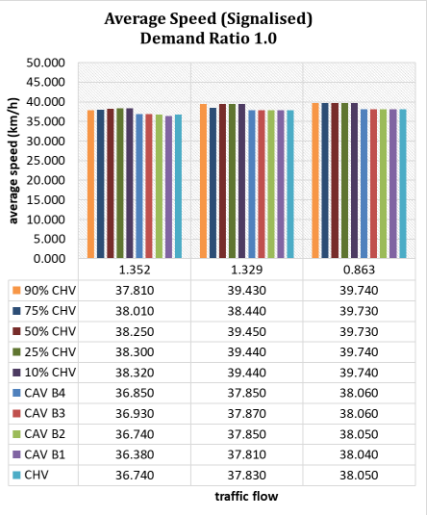
C3. Simulation Results on Average Vehicle Speed



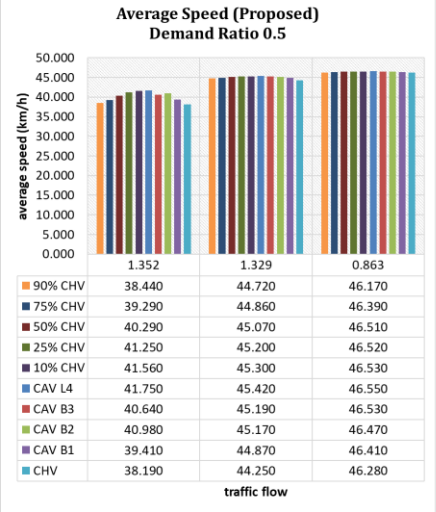
	1.352	1.329	0.863
90% CHV	37.860	44.460	45.270
75% CHV	38.570	44.500	45.900
50% CHV	39.530	44.700	46.310
25% CHV	40.820	44.790	46.450
10% CHV	41.080	44.920	46.480
CAV L4	41.690	45.090	46.450
CAV B3	40.330	44.950	46.370
CAV B2	39.390	44.610	46.360
CAV B1	38.810	44.240	46.210
CHV	37.520	44.090	46.180



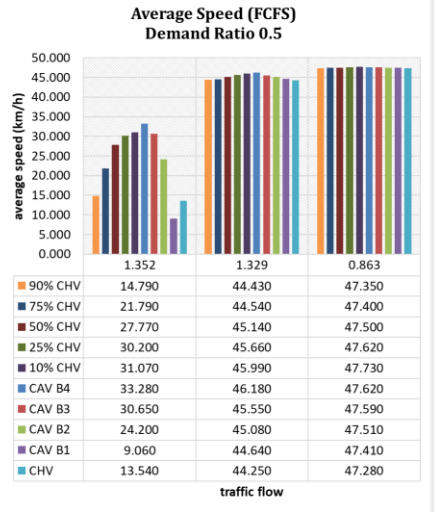
	1.352	1.329	0.863
90% CHV	24.340	42.720	47.060
75% CHV	26.790	42.690	47.120
50% CHV	28.920	43.700	47.280
25% CHV	31.270	44.530	47.420
10% CHV	32.740	45.010	47.570
CAV B4	33.360	45.300	47.620
CAV B3	30.330	44.380	47.390
CAV B2	25.250	43.730	47.280
CAV B1	24.040	42.830	47.160
CHV	23.570	42.430	47.010



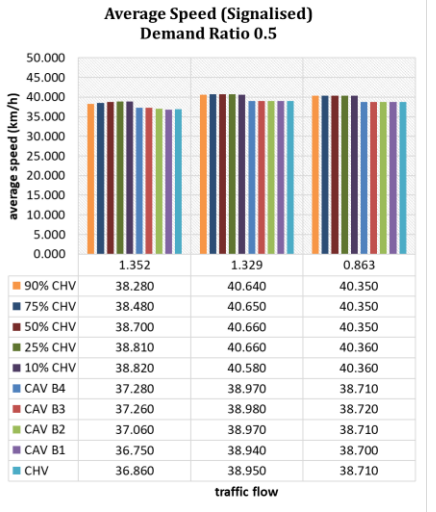
	1.352	1.329	0.863
90% CHV	37.810	39.430	39.740
75% CHV	38.010	38.440	39.730
50% CHV	38.250	39.450	39.730
25% CHV	38.300	39.440	39.740
10% CHV	38.320	39.440	39.740
CAV B4	36.850	37.850	38.060
CAV B3	36.930	37.870	38.060
CAV B2	36.740	37.850	38.050
CAV B1	36.380	37.810	38.040
CHV	36.740	37.830	38.050



	1.352	1.329	0.863
90% CHV	38.440	44.720	46.170
75% CHV	39.290	44.860	46.390
50% CHV	40.290	45.070	46.510
25% CHV	41.250	45.200	46.520
10% CHV	41.560	45.300	46.530
CAV L4	41.750	45.420	46.550
CAV B3	40.640	45.190	46.530
CAV B2	40.980	45.170	46.470
CAV B1	39.410	44.870	46.410
CHV	38.190	44.250	46.280

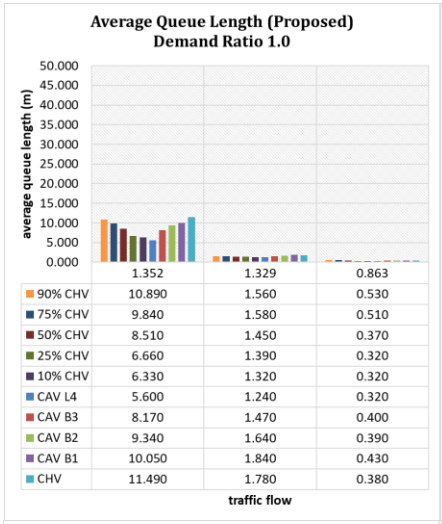


	1.352	1.329	0.863
90% CHV	14.790	44.430	47.350
75% CHV	21.790	44.540	47.400
50% CHV	27.770	45.140	47.500
25% CHV	30.200	45.660	47.620
10% CHV	31.070	45.990	47.730
CAV B4	33.280	46.180	47.620
CAV B3	30.650	45.550	47.590
CAV B2	24.200	45.080	47.510
CAV B1	9.060	44.640	47.410
CHV	13.540	44.250	47.280



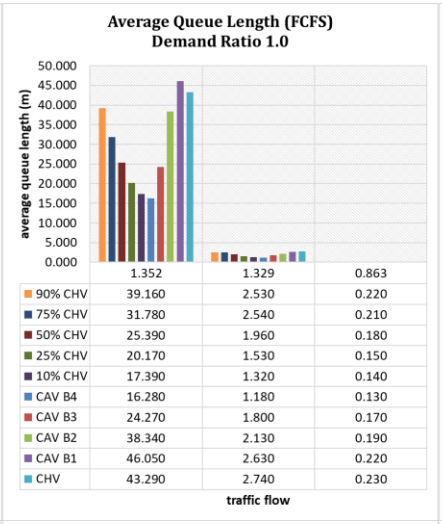
	1.352	1.329	0.863
90% CHV	38.280	40.640	40.350
75% CHV	38.480	40.650	40.350
50% CHV	38.700	40.660	40.350
25% CHV	38.810	40.660	40.360
10% CHV	38.820	40.580	40.360
CAV B4	37.280	38.970	38.710
CAV B3	37.260	38.980	38.720
CAV B2	37.060	38.970	38.710
CAV B1	36.750	38.940	38.700
CHV	36.860	38.950	38.710

C4. Simulation Results on Average Queue Length



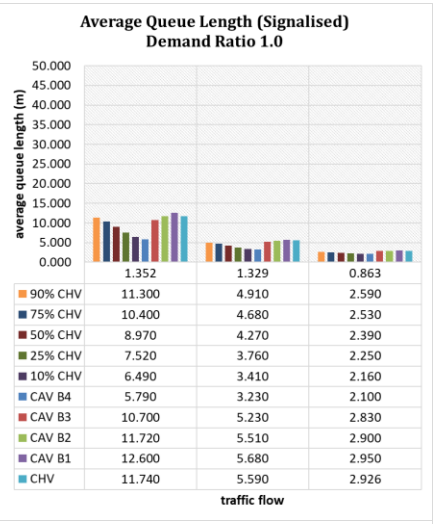
	1.352	1.329	0.863
90% CHV	10.890	1.560	0.530
75% CHV	9.840	1.580	0.510
50% CHV	8.510	1.450	0.370
25% CHV	6.660	1.390	0.320
10% CHV	6.330	1.320	0.320
CAV L4	5.600	1.240	0.320
CAV B3	8.170	1.470	0.400
CAV B2	9.340	1.640	0.390
CAV B1	10.050	1.840	0.430
CHV	11.490	1.780	0.380

traffic flow



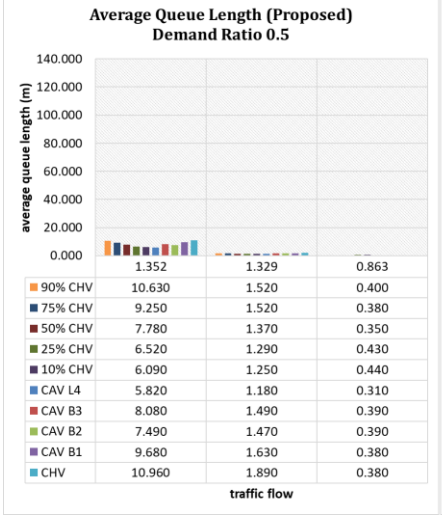
	1.352	1.329	0.863
90% CHV	39.160	2.530	0.220
75% CHV	31.780	2.540	0.210
50% CHV	25.390	1.960	0.180
25% CHV	20.170	1.530	0.150
10% CHV	17.390	1.320	0.140
CAV B4	16.280	1.180	0.130
CAV B3	24.270	1.800	0.170
CAV B2	38.340	2.130	0.190
CAV B1	46.050	2.630	0.220
CHV	43.290	2.740	0.230

traffic flow



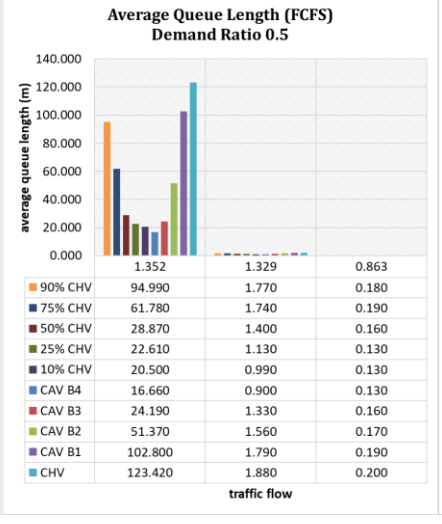
	1.352	1.329	0.863
90% CHV	11.300	4.910	2.590
75% CHV	10.400	4.680	2.530
50% CHV	8.970	4.270	2.390
25% CHV	7.520	3.760	2.250
10% CHV	6.490	3.410	2.160
CAV B4	5.790	3.230	2.100
CAV B3	10.700	5.230	2.830
CAV B2	11.720	5.510	2.900
CAV B1	12.600	5.680	2.950
CHV	11.740	5.590	2.926

traffic flow



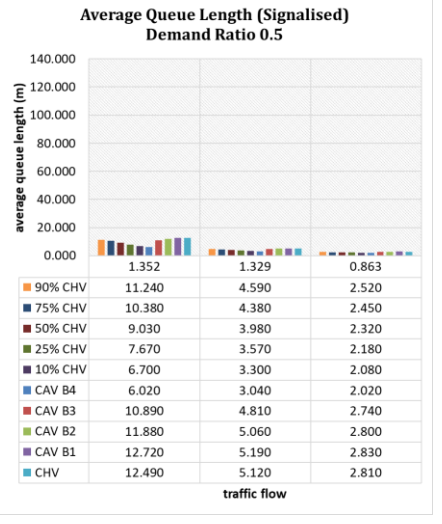
	1.352	1.329	0.863
90% CHV	10.630	1.520	0.400
75% CHV	9.250	1.520	0.380
50% CHV	7.780	1.370	0.350
25% CHV	6.520	1.290	0.430
10% CHV	6.090	1.250	0.440
CAV L4	5.820	1.180	0.310
CAV B3	8.080	1.490	0.390
CAV B2	7.490	1.470	0.390
CAV B1	9.680	1.630	0.380
CHV	10.960	1.890	0.380

traffic flow



	1.352	1.329	0.863
90% CHV	94.990	1.770	0.180
75% CHV	61.780	1.740	0.190
50% CHV	28.870	1.400	0.160
25% CHV	22.610	1.130	0.130
10% CHV	20.500	0.990	0.130
CAV B4	16.660	0.900	0.130
CAV B3	24.190	1.330	0.160
CAV B2	51.370	1.560	0.170
CAV B1	102.800	1.790	0.190
CHV	123.420	1.880	0.200

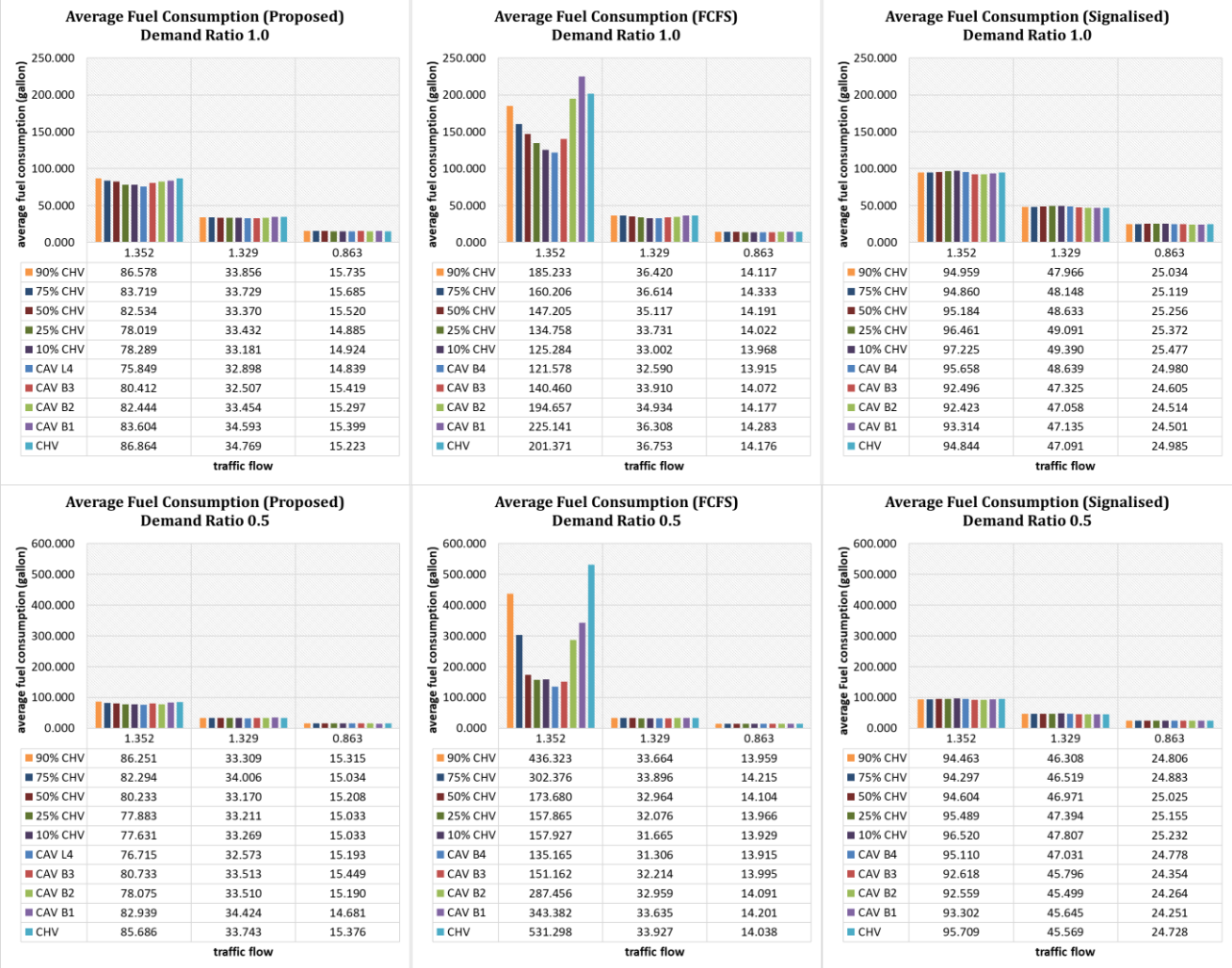
traffic flow



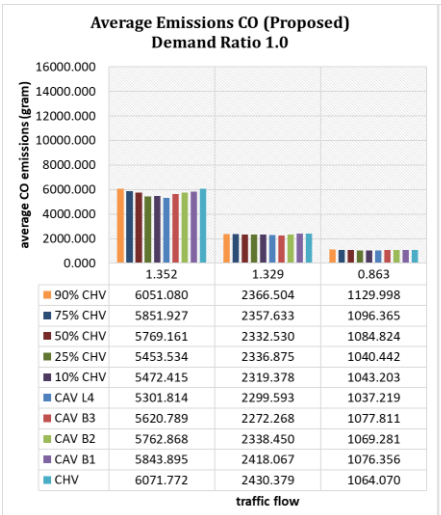
	1.352	1.329	0.863
90% CHV	11.240	4.590	2.520
75% CHV	10.380	4.380	2.450
50% CHV	9.030	3.980	2.320
25% CHV	7.670	3.570	2.180
10% CHV	6.700	3.300	2.080
CAV B4	6.020	3.040	2.020
CAV B3	10.890	4.810	2.740
CAV B2	11.880	5.060	2.800
CAV B1	12.720	5.190	2.830
CHV	12.490	5.120	2.810

traffic flow

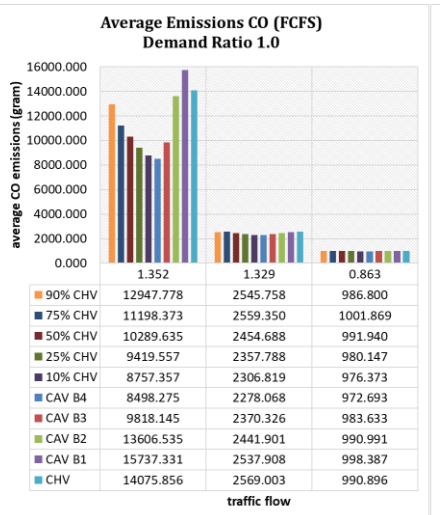
C5. Simulation Results on Average Fuel Consumption



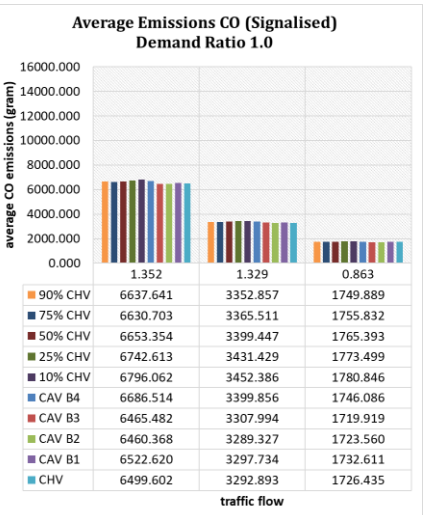
C6. Simulation Results on Average CO Emissions



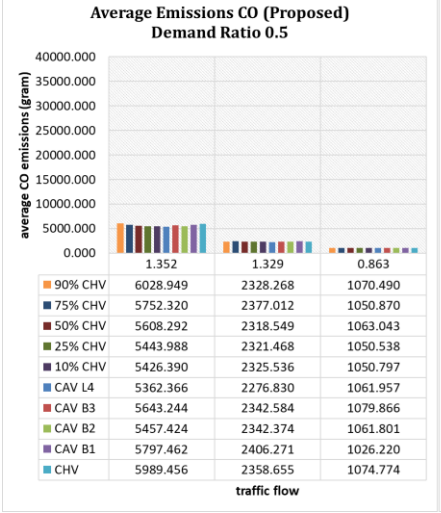
	1.352	1.329	0.863
90% CHV	6051.080	2366.504	1129.998
75% CHV	5851.927	2357.633	1096.365
50% CHV	5769.161	2332.530	1084.824
25% CHV	5453.534	2336.875	1040.442
10% CHV	5472.415	2319.378	1043.203
CAV L4	5301.814	2299.593	1037.219
CAV B3	5620.789	2272.268	1077.811
CAV B2	5762.868	2338.450	1069.281
CAV B1	5843.895	2418.067	1076.356
CHV	6071.772	2430.379	1064.070



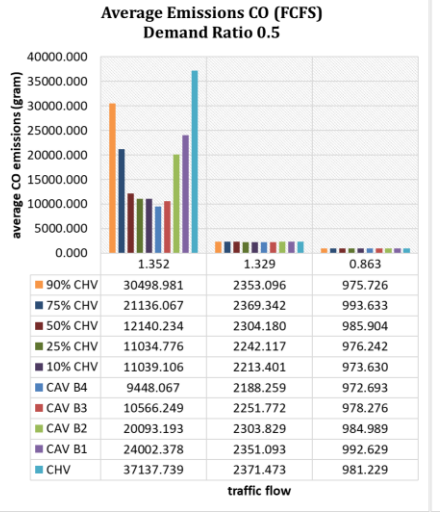
	1.352	1.329	0.863
90% CHV	12947.778	2545.758	986.800
75% CHV	11198.373	2559.350	1001.869
50% CHV	10289.635	2454.688	991.940
25% CHV	9419.557	2357.788	980.147
10% CHV	8757.357	2306.819	976.373
CAV B4	8498.275	2278.068	972.693
CAV B3	9818.145	2370.326	983.633
CAV B2	13606.535	2441.901	990.991
CAV B1	15737.331	2537.908	998.387
CHV	14075.856	2569.003	990.896



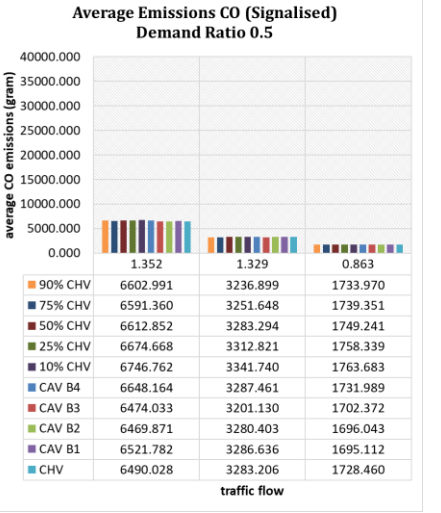
	1.352	1.329	0.863
90% CHV	6637.641	3352.857	1749.889
75% CHV	6630.703	3365.511	1755.832
50% CHV	6653.354	3399.447	1765.393
25% CHV	6742.613	3431.429	1773.499
10% CHV	6796.062	3452.386	1780.846
CAV B4	6686.514	3399.856	1746.086
CAV B3	6465.482	3307.994	1719.919
CAV B2	6460.368	3289.327	1723.560
CAV B1	6522.620	3297.734	1732.611
CHV	6499.602	3292.893	1726.435



	1.352	1.329	0.863
90% CHV	6028.949	2328.268	1070.490
75% CHV	5752.320	2377.012	1050.870
50% CHV	5608.292	2318.549	1063.043
25% CHV	5443.988	2321.468	1050.538
10% CHV	5426.390	2325.536	1050.797
CAV L4	5362.366	2276.830	1061.957
CAV B3	5643.244	2342.584	1079.866
CAV B2	5457.424	2342.374	1061.801
CAV B1	5797.462	2406.271	1026.220
CHV	5989.456	2358.655	1074.774

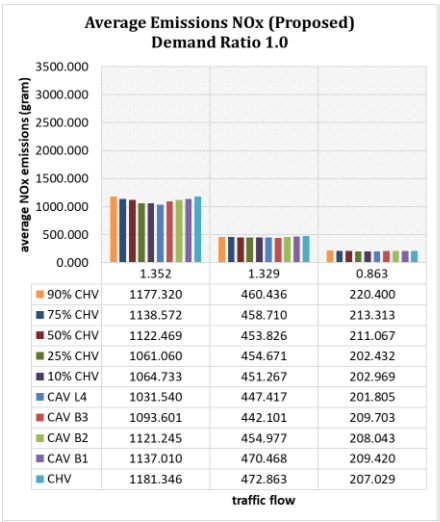


	1.352	1.329	0.863
90% CHV	30498.981	2353.096	975.726
75% CHV	21136.067	2369.342	993.633
50% CHV	12140.234	2304.180	985.904
25% CHV	11034.776	2242.117	976.242
10% CHV	11039.106	2213.401	973.630
CAV B4	9448.067	2188.259	972.693
CAV B3	10566.249	2251.772	978.276
CAV B2	20093.193	2303.829	984.989
CAV B1	24002.378	2351.093	992.629
CHV	37137.739	2371.473	981.229

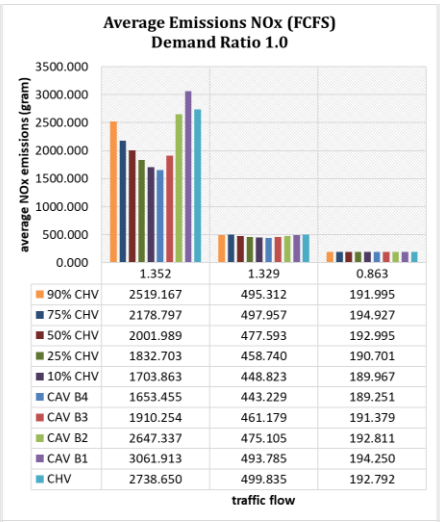


	1.352	1.329	0.863
90% CHV	6602.991	3236.899	1733.970
75% CHV	6591.360	3251.648	1739.351
50% CHV	6612.852	3283.294	1749.241
25% CHV	6674.668	3312.821	1758.339
10% CHV	6746.762	3341.740	1763.683
CAV B4	6648.164	3287.461	1731.989
CAV B3	6474.033	3201.130	1702.372
CAV B2	6469.871	3280.403	1696.043
CAV B1	6521.782	3286.636	1695.112
CHV	6490.028	3283.206	1728.460

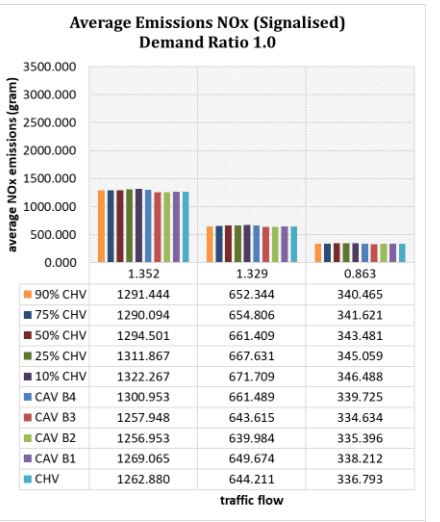
C7. Simulation Results on Average NOx Emissions



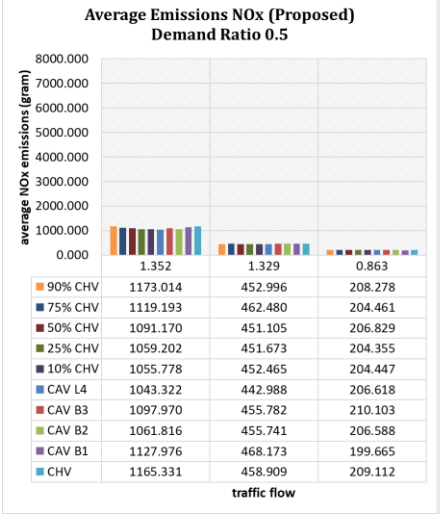
	1.352	1.329	0.863
90% CHV	1177.320	460.436	220.400
75% CHV	1138.572	458.710	213.313
50% CHV	1122.469	453.826	211.067
25% CHV	1061.060	454.671	202.432
10% CHV	1064.733	451.267	202.969
CAV L4	1031.540	447.417	201.805
CAV B3	1093.601	442.101	209.703
CAV B2	1121.245	454.977	208.043
CAV B1	1137.010	470.468	209.420
CHV	1181.346	472.863	207.029



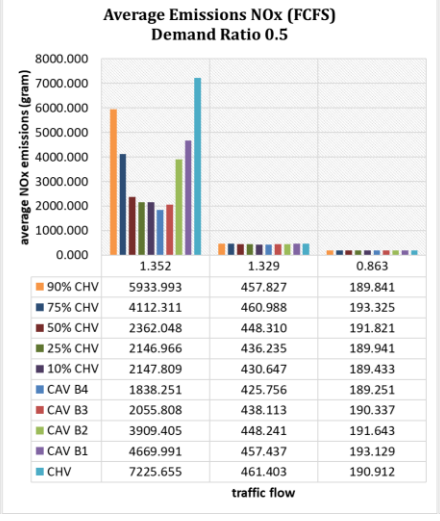
	1.352	1.329	0.863
90% CHV	2519.167	495.312	191.995
75% CHV	2178.797	497.957	194.927
50% CHV	2001.989	477.593	192.995
25% CHV	1832.703	458.740	190.701
10% CHV	1703.863	448.823	189.967
CAV B4	1653.455	443.229	189.251
CAV B3	1910.254	461.179	191.379
CAV B2	2647.337	475.105	192.811
CAV B1	3061.913	493.785	194.250
CHV	2738.650	499.835	192.792



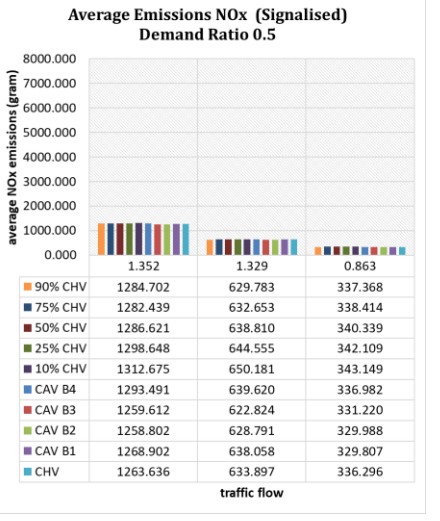
	1.352	1.329	0.863
90% CHV	1291.444	652.344	340.465
75% CHV	1290.094	654.806	341.621
50% CHV	1294.501	661.409	343.481
25% CHV	1311.867	667.631	345.059
10% CHV	1322.267	671.709	346.488
CAV B4	1300.953	661.489	339.725
CAV B3	1257.948	643.615	334.634
CAV B2	1256.953	639.984	335.396
CAV B1	1269.065	649.674	338.212
CHV	1262.880	644.211	336.793



	1.352	1.329	0.863
90% CHV	1173.014	452.996	208.278
75% CHV	1119.193	462.480	204.461
50% CHV	1091.170	451.105	206.829
25% CHV	1059.202	451.673	204.355
10% CHV	1055.778	452.465	204.447
CAV L4	1043.322	442.988	206.618
CAV B3	1097.970	455.782	210.103
CAV B2	1061.816	455.741	206.588
CAV B1	1127.976	468.173	199.665
CHV	1165.331	458.909	209.112

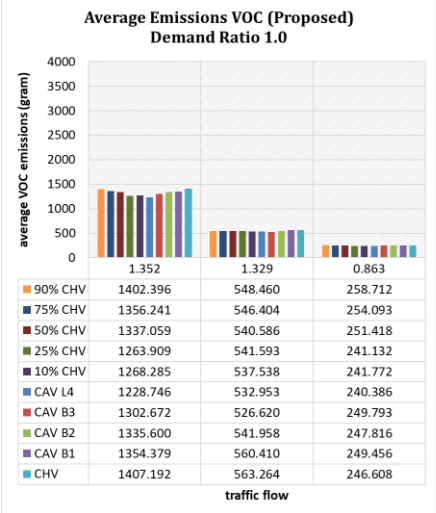


	1.352	1.329	0.863
90% CHV	5933.993	457.827	189.841
75% CHV	4112.311	460.988	193.325
50% CHV	2362.048	448.310	191.821
25% CHV	2146.966	436.235	189.941
10% CHV	2147.809	430.647	189.433
CAV B4	1838.251	425.756	189.251
CAV B3	2055.808	438.113	190.337
CAV B2	3909.405	448.241	191.643
CAV B1	4669.991	457.437	193.129
CHV	7225.655	461.403	190.912

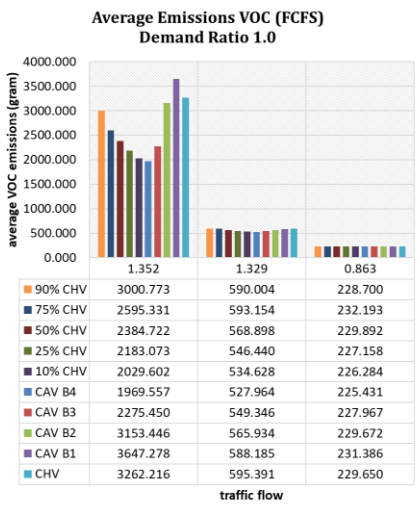


	1.352	1.329	0.863
90% CHV	1284.702	629.783	337.368
75% CHV	1282.439	632.653	338.414
50% CHV	1286.621	638.810	340.339
25% CHV	1298.648	644.555	342.109
10% CHV	1312.675	650.181	343.149
CAV B4	1293.491	639.620	336.982
CAV B3	1259.612	622.824	331.220
CAV B2	1258.802	628.791	329.988
CAV B1	1268.902	638.058	329.807
CHV	1263.636	633.897	336.296

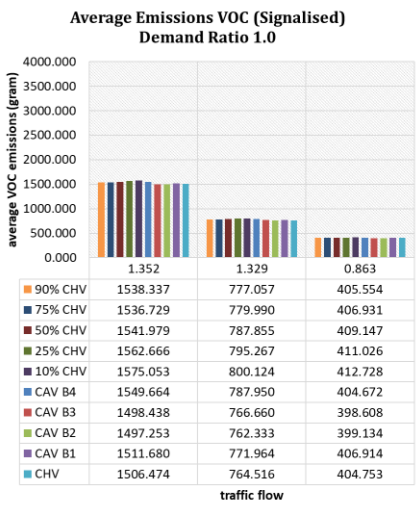
C8. Simulation Results on Average VOC Emissions



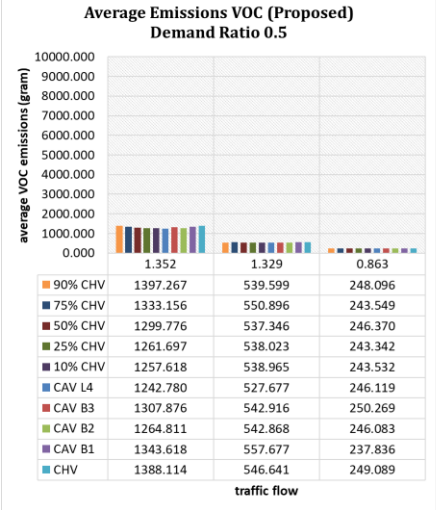
	1.352	1.329	0.863
90% CHV	1402.396	548.460	258.712
75% CHV	1356.241	546.404	254.093
50% CHV	1337.059	540.586	251.418
25% CHV	1263.909	541.593	241.132
10% CHV	1268.285	537.538	241.772
CAV L4	1228.746	532.953	240.386
CAV B3	1302.672	526.620	249.793
CAV B2	1335.600	541.958	247.816
CAV B1	1354.379	560.410	249.456
CHV	1407.192	563.264	246.608



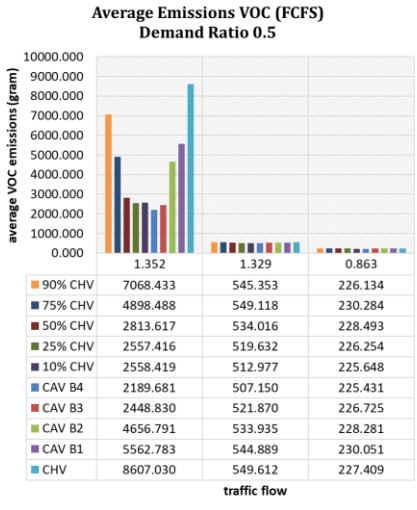
	1.352	1.329	0.863
90% CHV	3000.773	590.004	228.700
75% CHV	2595.331	593.154	232.193
50% CHV	2384.722	568.898	229.892
25% CHV	2183.073	546.440	227.158
10% CHV	2029.602	534.628	226.284
CAV B4	1969.557	527.964	225.431
CAV B3	2275.450	549.346	227.967
CAV B2	3153.446	565.934	229.672
CAV B1	3647.278	588.185	231.386
CHV	3262.216	595.391	229.650



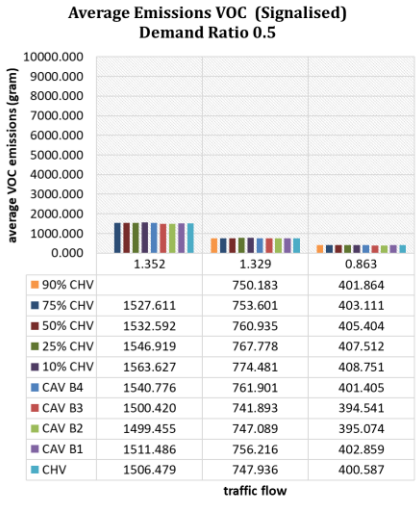
	1.352	1.329	0.863
90% CHV	1538.337	777.057	405.554
75% CHV	1536.729	779.990	406.931
50% CHV	1541.979	787.855	409.147
25% CHV	1562.666	795.267	411.026
10% CHV	1575.053	800.124	412.728
CAV B4	1549.664	787.950	404.672
CAV B3	1498.438	766.660	398.608
CAV B2	1497.253	762.333	399.134
CAV B1	1511.680	771.964	406.914
CHV	1506.474	764.516	404.753



	1.352	1.329	0.863
90% CHV	1397.267	539.599	248.096
75% CHV	1333.156	550.896	243.549
50% CHV	1299.776	537.346	246.370
25% CHV	1261.697	538.023	243.342
10% CHV	1257.618	538.965	243.532
CAV L4	1242.780	527.677	246.119
CAV B3	1307.876	542.916	250.269
CAV B2	1264.811	542.868	246.083
CAV B1	1343.618	557.677	237.836
CHV	1388.114	546.641	249.089



	1.352	1.329	0.863
90% CHV	7068.433	545.353	226.134
75% CHV	4898.488	549.118	230.284
50% CHV	2813.617	534.016	228.493
25% CHV	2557.416	519.632	226.254
10% CHV	2558.419	512.977	225.648
CAV B4	2189.681	507.150	225.431
CAV B3	2448.830	521.870	226.725
CAV B2	4656.791	533.935	228.281
CAV B1	5562.783	544.889	230.051
CHV	8607.030	549.612	227.409



	1.352	1.329	0.863
90% CHV		750.183	401.864
75% CHV	1527.611	753.601	403.111
50% CHV	1532.592	760.935	405.404
25% CHV	1546.919	767.778	407.512
10% CHV	1563.627	774.481	408.751
CAV B4	1540.776	761.901	401.405
CAV B3	1500.420	741.893	394.541
CAV B2	1499.455	747.089	395.074
CAV B1	1511.486	756.216	402.859
CHV	1506.479	747.936	400.587

Appendix D

D1. Published Work

The full text view version of the published paper can be accessed by using the link below:

https://link.springer.com/epdf/10.1007/s10470-018-1152-2?author_access_token=w0WJFa4wUghOauFDxMNEw_e4RwlQNchNByi7wbcMAY6E-rJOAtBoilRadzYbo0PfAsS545Dd0gTtUUIRRIpRPfziNttrGoa7_gN2D3V4dGrh9s6bF3nz7NfgHTclLCq3k9l1Pru6PKOhEXpaU3QpwA%3D%3D

References

Abadi, M. *et al.* (2016) 'TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems', *arXiv preprint arXiv:1603.04467*. Available at: http://arxiv.org/abs/1603.04467.

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: 10.1061/(ASCE)0733-947X(2003)129:3(278).

Ahn, H. *et al.* (2015) 'Experimental testing of a semi-autonomous multi-vehicle collision avoidance algorithm at an intersection testbed', in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Hamburg, Germany: IEEE, pp. 4834–4839. doi: 10.1109/IROS.2015.7354056.

Ahn, H. and Del Vecchio, D. (2016) 'Semi-autonomous Intersection Collision Avoidance through Job-shop Scheduling', in *Proceedings of the 19th International Conference on Hybrid Systems: Computation and Control - HSCC '16*. New York, New York, USA: ACM Press, pp. 185–194. doi: 10.1145/2883817.2883830.

Altche, F. and de La Fortelle, A. (2016) 'Analysis of optimal solutions to robot coordination problems to improve autonomous intersection management policies', in *2016 IEEE Intelligent Vehicles Symposium (IV)*. Gothenburg, Sweden: IEEE, pp. 86–91. doi: 10.1109/IVS.2016.7535369.

Altche, F., Qian, X. and de La Fortelle, A. (2016) 'Least restrictive and minimally deviating supervisor for Safe semi-autonomous driving at an intersection: An MIQP approach', in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. Rio de Janeiro, Brazil: IEEE, pp. 2520–2526. doi: 10.1109/ITSC.2016.7795961.

Arel, I. *et al.* (2010) 'Reinforcement learning-based multi-agent system for network traffic signal control', *IET Intelligent Transport Systems*, 4(2), p. 128. doi: 10.1049/iet-its.2009.0070.

Asselin-Miller, N. *et al.* (2016) *Study on the Deployment of C-ITS in Europe : Final Report*. London, UK. Available at: https://ec.europa.eu/transport/sites/transport/files/2016-c-its-deployment-study-final-report.pdf (Accessed: 27 January 2021).

Atkins (2016a) *Research on the impacts of connected and autonomous vehicles (CAVs) on traffic flow Stage 1: Evidence Review*. Available at: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/530091/impacts-of-connected-and-autonomous-vehicles-on-traffic-flow-summary-report.pdf (Accessed: 27 January 2021).

Atkins (2016b) *Research on the impacts of connected and autonomous vehicles (CAVs) on traffic flow Stage 2: Traffic Modelling and Analysis Technical Report*. Available at: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/530091/impacts-of-connected-and-autonomous-vehicles-on-traffic-flow-summary-report.pdf (Accessed: 12 March 2017).

Atkins (2016c) *Research on the impacts of connected and autonomous vehicles (CAVs) on traffic flow Summary Report*. Available at: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/530091/impacts-of-connected-and-autonomous-vehicles-on-traffic-flow-summary-report.pdf (Accessed: 27 January 2021).

Bakker, B. *et al.* (2010) 'Traffic Light Control by Multiagent Reinforcement Learning Systems', in *Interactive Collaborative Information Systems*, pp. 475–510. doi: 10.1007/978-3-642-11688-9_18.

Bishop, C. M. (1995) *Neural networks for pattern recognition*. Oxford university press.

Boillot, F., Midenet, S. and Pierrelée, J.-C. (2006) 'The real-time urban traffic control system CRONOS: Algorithm and experiments', *Transportation Research Part C: Emerging Technologies*, 14(1), pp. 18–38. doi: 10.1016/j.trc.2006.05.001.

Bojarski, M. *et al.* (2016) 'End to End Learning for Self-Driving Cars'. Available at: http://arxiv.org/abs/1604.07316 (Accessed: 27 January 2021).

British Standards Institution (2007) *BS 6100-2: Building and civil engineering. Vocabulary. Spaces, building types, environment and physical planning*. Available at:

https://shop.bsigroup.com/ProductDetail/?pid=000000000030169336 (Accessed: 27 January 2021).

Cai, B. *et al.* (2014) 'Unsignalized cooperative optimization control method based on vehicle speed guidance and information interaction', in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Qingdao, China: IEEE, pp. 57–62. doi: 10.1109/ITSC.2014.6957666.

Cai, C., Wang, Y. and Geers, G. (2010) 'Adaptive traffic signal control using vehicle-to-infrastructure communication', in *Proceedings of the Second International Workshop on Computational Transportation Science - IWCTS '10*. New York, New York, USA: ACM Press, p. 43. doi: 10.1145/1899441.1899453.

Camacho, E. F. and Bordons, C. (2007) *Model Predictive control*. London: Springer London (Advanced Textbooks in Control and Signal Processing). doi: 10.1007/978-0-85729-398-5.

de Campos, G. R., Falcone, P. and Sjoberg, J. (2013) 'Autonomous cooperative driving: A velocity-based negotiation approach for intersection crossing', in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. The Hague, The Netherlands: IEEE, pp. 1456–1461. doi: 10.1109/ITSC.2013.6728435.

Carlino, D., Boyles, S. D. and Stone, P. (2013) 'Auction-based autonomous intersection management', in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. The Hague, The Netherlands: IEEE, pp. 529–534. doi: 10.1109/ITSC.2013.6728285.

Casas, N. (2017) 'Deep Deterministic Policy Gradient for Urban Traffic Light Control', *arXiv preprint*, pp. 1–12. Available at: http://arxiv.org/abs/1703.09035 (Accessed: 27 January 2021).

Chang, H.-J. and Park, G.-T. (2013) 'A study on traffic signal control at signalized intersections in vehicular ad hoc networks', *Ad Hoc Networks*. Elsevier B.V., 11(7), pp. 2115–2124. doi: 10.1016/j.adhoc.2012.02.013.

Chen, G. and Kang, K.-D. (2015) 'Win-fit: Efficient intersection management via dynamic vehicle batching and scheduling', in *2015 International Conference on Connected Vehicles and Expo (ICCVE)*. IEEE, pp. 263–270. doi: 10.1109/ICCVE.2015.17.

Cheng, J. *et al.* (2017) 'Fuzzy Group-Based Intersection Control via Vehicular Networks for Smart Transportations', *IEEE Transactions on Industrial Informatics*, 13(2), pp. 751–758. doi: 10.1109/TII.2016.2590302.

Curry, H. B. (1944) 'The method of steepest descent for non-linear minimization problems', *Quarterly of Applied Mathematics*, 2(3), pp. 258–261.

Dai, P. *et al.* (2016) 'Quality-of-Experience-Oriented Autonomous Intersection Control in Vehicular Networks', *IEEE Transactions on Intelligent Transportation Systems*, 17(7), pp. 1956–1967. doi: 10.1109/TITS.2016.2514271.

Dinopoulou, V., Diakaki, C. and Papageorgiou, M. (2006) 'Applications of the urban traffic control strategy TUC', *European Journal of Operational Research*, 175(3), pp. 1652–1665. doi: 10.1016/j.ejor.2005.02.032.

Donkeycar (2021) *Donkeycar*. Available at: http://docs.donkeycar.com/guide/build_hardware/ (Accessed: 27 January 2021).

Dresner, K. and Stone, P. (2004) 'Multiagent Traffic Management: A Reservation-Based Intersection Control Mechanism', in *AAMAS '04 Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems - Volume 2*. Washington: IEEE Computer Society, pp. 530–537. doi: 10.1109/AAMAS.2004.190.

Dresner, K. and Stone, P. (2005) 'Turning the corner: improved intersection control for autonomous vehicles', in *IEEE Proceedings. Intelligent Vehicles Symposium, 2005.* IEEE, pp. 423–428. doi: 10.1109/IVS.2005.1505140.

Dresner, K. and Stone, P. (2008) 'A Multiagent Approach to Autonomous Intersection Management', *Journal of Artificial Intelligence Research (JAIR)*, 31(1), pp. 591–656.

Du, Z., HomChaudhuri, B. and Pisu, P. (2017) 'Coordination strategy for vehicles passing multiple signalized intersections: A connected vehicle penetration rate study', in *2017 American Control Conference (ACC)*. Seattle, USA: IEEE, pp. 4952–4957. doi: 10.23919/ACC.2017.7963722.

El-Tantawy, S. and Abdulhai, B. (2010) 'An agent-based learning towards decentralized and coordinated traffic

signal control', in *13th International IEEE Conference on Intelligent Transportation Systems*. Madeira Island, Portugal: IEEE, pp. 665–670. doi: 10.1109/ITSC.2010.5625066.

Elhenawy, M. *et al.* (2015) 'An Intersection Game-Theory-Based Traffic Control Algorithm in a Connected Vehicle Environment', in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. IEEE, pp. 343–347. doi: 10.1109/ITSC.2015.65.

Englund, C. *et al.* (2016) 'The Grand Cooperative Driving Challenge 2016: boosting the introduction of cooperative automated vehicles', *IEEE Wireless Communications*, 23(4), pp. 146–152. doi: 10.1109/MWC.2016.7553038.

ETSI (2010a) *ES 202 663 - V1.1.0 - Intelligent Transport Systems (ITS); European profile standard for the physical and medium access control layer of Intelligent Transport Systems operating in the 5 GHz frequency band*, *ETSI*. Cedex, France. Available at: https://standards.iteh.ai/catalog/standards/etsi/88813933-a28b-4cad-9faa-3a06ccc012c6/etsi-es-202-663-v1.1.0-2010-01 (Accessed: 27 January 2021).

ETSI (2010b) *ETSI EN 302 665 Intelligent Transport Systems (ITS); Communications Architecture*, *ETSI*. ETSI. Available at: http://www.etsi.org/deliver/etsi_en/302600_302699/302665/01.01.01_60/en_302665v010101p.pdf (Accessed: 27 January 2021).

ETSI (2013) *ETSI TS 101 539-3 Intelligent Transport Systems (ITS); V2X Applications; Part 3: Longitudinal Collision Risk Warning (LCRW) application requirements specification*, *ETSI*. ETSI. Available at: http://www.etsi.org/deliver/etsi_ts/101500_101599/10153903/01.01.01_60/ts_10153903v010101p.pdf (Accessed: 27 January 2021).

ETSI (2014a) *ETSI EN 302 637-2 V1.3.2 Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*, *ETSI*. ETSI. Available at: http://www.etsi.org/deliver/etsi_en/302600_302699/30263702/01.03.02_60/en_30263702v010302p.pdf (Accessed: 27 January 2021).

ETSI (2014b) *ETSI EN 302 637-3 V1.2.2 Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service*, *ETSI*. ETSI. Available at: http://www.etsi.org/deliver/etsi_en/302600_302699/30263703/01.02.02_60/en_30263703v010202p.pdf (Accessed: 27 January 2021).

ETSI (2014c) *ETSI TS 102 894-2 V1.2.1 Intelligent Transport Systems (ITS); Users and applications requirements; Part 2: Applications and facilities layer common data dictionary*, *ETSI*. ETSI. Available at: http://www.etsi.org/deliver/etsi_ts/102800_102899/10289402/01.02.01_60/ts_10289402v010201p.pdf (Accessed: 27 January 2021).

European Commision (2016) *COM (2016) 766 final [C-ITS]*. Available at: https://eur-lex.europa.eu/legal-content/en/ALL/?uri=CELEX:52016DC0766 (Accessed: 27 January 2021).

European Commission (2017) *C-ITS Platform: Phase II Final Report*, *European Commission*. Brussels, Belgium. Available at: https://ec.europa.eu/transport/sites/transport/files/2017-09-c-its-platform-final-report.pdf (Accessed: 27 January 2021).

European Commission (2018) *Traffic Safety Basic Facts on Junctions 2018*. Available at: https://ec.europa.eu/transport/road_safety/sites/roadsafety/files/pdf/statistics/dacota/bfs2018_junctions.pdf (Accessed: 27 January 2021).

European Commission (2019a) *Statistical Pocketbook 2019: EU transport in figures*. Brussels, Belgium. doi: 10.2832/729667.

European Commission (2019b) *Transport in the European Union - Current Trends and Issues*, *European Commission*. Brussels, Belgium. Available at: https://ec.europa.eu/transport/sites/transport/files/2019-transport-in-the-eu-current-trends-and-issues.pdf (Accessed: 27 January 2021).

Fagnant, D. J. and Kockelman, K. (2015) 'Preparing a nation for autonomous vehicles: opportunities, barriers and policy recommendations', *Transportation Research Part A: Policy and Practice*. Elsevier Ltd, 77, pp. 167–181. doi: 10.1016/j.tra.2015.04.003.

Fayazi, S. A., Vahidi, A. and Luckow, A. (2017) 'Optimal scheduling of autonomous vehicle arrivals at intelligent intersections via MILP', in *2017 American Control Conference (ACC)*. Seattle, USA: IEEE, pp.

4920–4925. doi: 10.23919/ACC.2017.7963717.

FHWA (2001) *Traffic Flow Theory A State-of-the-Art Report*, *TRB Special Report 165 - Revised Monograph on Traffic Flow Theory*. Available at: https://www.researchgate.net/publication/248146380_Traffic_Flow_Theory_A_State-of-the-Art_Report (Accessed: 27 January 2021).

FHWA (2004) *Signalized Intersections: Informational Guide*. Available at: https://www.fhwa.dot.gov/publications/research/safety/04091/04091.pdf (Accessed: 27 January 2021).

Fujimoto, S., van Hoof, H. and Meger, D. (2018) 'Addressing Function Approximation Error in Actor-Critic Methods', *arXiv preprint arXiv:1802.09477*. Available at: http://arxiv.org/abs/1802.09477.

Gao, J. *et al.* (2017) 'Adaptive Traffic Signal Control: Deep Reinforcement Learning Algorithm with Experience Replay and Target Network', *arXiv preprint arXiv:1705.02755*, pp. 1–10. Available at: http://arxiv.org/abs/1705.02755 (Accessed: 27 January 2021).

Garrido-Jurado, S. *et al.* (2014) 'Automatic generation and detection of highly reliable fiducial markers under occlusion', *Pattern Recognition*. Elsevier, 47(6), pp. 2280–2292.

Gartner, N. H. (1983) *OPAC: A demand-responsive strategy for traffic signal control*, *Transportation Research Record*. Washington D. C.: Transportation Research Board. Available at: https://trid.trb.org/view.aspx?id=196609 (Accessed: 27 January 2021).

Gartner, N., Pooran, F. and Andrews, C. (2002) 'Optimized Policies for Adaptive Control Strategy in Real-Time Traffic Adaptive Control Systems: Implementation and Field Testing', *Transportation Research Record: Journal of the Transportation Research Board*, 1811, pp. 148–156. doi: 10.3141/1811-18.

Gáspár, P., Szalay, Z. and Aradi, S. (2014) 'Highly Automated Vehicle Systems', (October), p. 187. Available at: http://www.mogi.bme.hu/TAMOP/jarmurendszerek_iranyitasa_angol/index.html.

Ghaffarian, H., Fathy, M. and Soryani, M. (2012) 'Vehicular ad hoc networks enabled traffic controller for removing traffic lights in isolated intersections based on integer linear programming', *IET Intelligent Transport Systems*, 6(2), p. 115. doi: 10.1049/iet-its.2010.0207.

Gokulan, B. P. and Srinivasan, D. (2010) 'Distributed Geometric Fuzzy Multiagent Urban Traffic Signal Control', *IEEE Transactions on Intelligent Transportation Systems*, 11(3), pp. 714–727. doi: 10.1109/TITS.2010.2050688.

Gregoire, J., Bonnabel, S. and de La Fortelle, A. (2013) 'Optimal cooperative motion planning for vehicles at intersections', *IEEE IV 2012 WORKSHOP ON NAVIGATION, ACCURATE POSITIONING AND MAPPING FOR INTELLIGEN.. 2012*. Available at: http://arxiv.org/abs/1310.7729 (Accessed: 27 January 2021).

Gregoire, J. and Frazzoli, E. (2016) 'Hybrid centralized/distributed autonomous intersection control: Using a job scheduler as a planner and inheriting its efficiency guarantees', in *2016 IEEE 55th Conference on Decision and Control (CDC)*. Las Vegas, USA: IEEE, pp. 2549–2554. doi: 10.1109/CDC.2016.7798646.

Gritschneder, F. *et al.* (2016) 'Adaptive learning based on guided exploration for decision making at roundabouts', in *2016 IEEE Intelligent Vehicles Symposium (IV)*. Gothenburg, Sweden: IEEE, pp. 433–440. doi: 10.1109/IVS.2016.7535422.

Haarnoja, T. *et al.* (2018) 'Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor', *arXiv preprint arXiv:1801.01290*. Available at: http://arxiv.org/abs/1801.01290.

Hafner, M. R. *et al.* (2013) 'Cooperative Collision Avoidance at Intersections: Algorithms and Experiments', *IEEE Transactions on Intelligent Transportation Systems*, 14(3), pp. 1162–1175. doi: 10.1109/TITS.2013.2252901.

Hassan, A. A. and Rakha, H. A. (2014) 'A Fully-Distributed Heuristic Algorithm for Control of Autonomous Vehicle Movements at Isolated Intersections', *International Journal of Transportation Science and Technology*. Tongji University and Tongji University Press, 3(4), pp. 297–309. doi: 10.1260/2046-0430.3.4.297.

Haydari, A. and Yilmaz, Y. (2020) 'Deep Reinforcement Learning for Intelligent Transportation Systems: A Survey', pp. 1–22. Available at: http://arxiv.org/abs/2005.00935.

Head, K. L., Mirchandani, P. B. and Sheppard, D. (1992) 'Hierarchical framework for real-time traffic control',

Transportation Research Record. Transportation Research Board, (1360), pp. 82–88. Available at: https://trid.trb.org/view.aspx?id=370960 (Accessed: 27 January 2021).

Henry, J. J., Farges, J. L. and Tuffal, J. (1984) 'The PRODYN real time traffic algorithm', in *Control in Transportation Systems*. Elsevier, pp. 305–310. doi: 10.1016/B978-0-08-029365-3.50048-1.

Hestenes, M. R. and Stiefel, E. (1952) 'Methods of conjugate gradients for solving linear systems', *Journal of research of the National Bureau of Standards*, 49(6), pp. 409–436.

Huang, S., Sadek, A. W. and Zhao, Y. (2012) 'Assessing the Mobility and Environmental Benefits of Reservation-Based Intelligent Intersections Using an Integrated Simulator', *IEEE Transactions on Intelligent Transportation Systems*, 13(3), pp. 1201–1214. doi: 10.1109/TITS.2012.2186442.

Hult, R. *et al.* (2015) 'An approximate solution to the optimal coordination problem for autonomous vehicles at intersections', in *2015 American Control Conference (ACC)*. Chicago, IL, USA: IEEE, pp. 763–768. doi: 10.1109/ACC.2015.7170826.

Hunt, P. B. *et al.* (1981) *SCOOT-a traffic responsive method of coordinating signals*, *TRRL Laboratory Report 1014*. Crowthorne, Berkshire: Department of the Environment Department of Transport. Available at: https://trl.co.uk/reports/LR1014 (Accessed: 27 January 2021).

Husch, D. and Albeck, J. (2003) *Synchro 6: Traffic Signal Software - User Guide*. Albany, CA.: Trafficware Ltd. Available at: https://books.google.co.uk/books/about/Synchro_6.html?id=W3EzPQAACAAJ&redir_esc=y (Accessed: 27 January 2021).

IEEE (2014) *IEEE Guide for Wireless Access in Vehicular Environments (WAVE) - Architecture*, *IEEE*. doi: 10.1109/IEEESTD.2014.6755433.

Itseez (2015) *Open Source Computer Vision Library*. Available at: https://github.com/itseez/opencv (Accessed: 27 January 2021).

Jin, Q. *et al.* (2012) 'Multi-Agent Intersection Management for Connected Vehicles Using an Optimal Scheduling Approach', in *2012 International Conference on Connected Vehicles and Expo (ICCVE)*. IEEE, pp. 185–190. doi: 10.1109/ICCVE.2012.41.

Katriniok, A. *et al.* (2017) 'Automation of Road Vehicles Using V2X: An Application to Intersection Automation', in *SAE Technical Paper 2017-01-0078*. SAE International. doi: 10.4271/2017-01-0078.

Katwijk, R. T. and Gabriel, S. (2015) 'Optimising a vehicle's approach towards an adaptively controlled intersection', *IET Intelligent Transport Systems*, 9(5), pp. 479–487. doi: 10.1049/iet-its.2014.0155.

Keong, C. K. (1993) 'The GLIDE system—Singapore's urban traffic control system', *Transport Reviews*, 13(4), pp. 295–305. doi: 10.1080/01441649308716854.

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*. Elsevier, 29, pp. 134–151. doi: 10.1016/j.engappai.2014.01.007.

Khayatian, M. *et al.* (2020) 'A Survey on Intersection Management of Connected Autonomous Vehicles', *ACM Transactions on Cyber-Physical Systems*, 4(4), pp. 1–27. doi: 10.1145/3407903.

Kingma, D. P. and Ba, J. (2014) 'Adam: A Method for Stochastic Optimization', *arXiv preprint arXiv:1412.6980*. Available at: http://arxiv.org/abs/1412.6980.

Kronborg, P. and Davidsson, F. (1993) 'MOVA and LHOVRA: Traffic signal control for isolated intersections', *Traffic engineering and control*, 34(4), pp. 195–200. Available at: https://trid.trb.org/view.aspx?id=516825 (Accessed: 27 January 2021).

Kronborg, P., Davidsson, F. and Edholm, J. (1997) *SOS - Self Optimising Signal Control: Development and Field Trials of the SOS Algorithm for Self Optimising Signal Control at Isolated Intersections*, *TFK - Transport Research Institute, Sweden*. Stockholm, Sweden: Transport Research Institute. Available at: http://www.durbit.se/Archives/ExternalPDF/Deliverables/PhaseII/Domain3/P2D_A31.3_SOS - self optimising signal control.pdf (Accessed: 27 January 2021).

Lam, S. and Katupitiya, J. (2013) 'Cooperative Intersection Negotiation for Multiple Autonomous Platoons', *IFAC Proceedings Volumes*. IFAC, 46(10), pp. 48–53. doi: 10.3182/20130626-3-AU-2035.00073.

Lee, J. *et al.* (2013) 'Sustainability assessments of cooperative vehicle intersection control at an urban corridor', *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd, 32, pp. 193–206. doi: 10.1016/j.trc.2012.09.004.

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.

Lee, J. and Park, B. B. (2015) 'Investigating communications performance for automated vehicle-based intersection control under connected vehicle environment', in *2015 IEEE Intelligent Vehicles Symposium (IV)*. Seoul, Korea: IEEE, pp. 1342–1347. doi: 10.1109/IVS.2015.7225902.

Levenberg, K. (1944) 'A method for the solution of certain non-linear problems in least squares', *Quarterly of applied mathematics*, 2(2), pp. 164–168.

Levin, M. W., Boyles, S. D. and Patel, R. (2016) 'Paradoxes of reservation-based intersection controls in traffic networks', *Transportation Research Part A: Policy and Practice*. Elsevier Ltd, 90, pp. 14–25. doi: 10.1016/j.tra.2016.05.013.

Levin, M. W. and Rey, D. (2017) 'Conflict-point formulation of intersection control for autonomous vehicles', *Transportation Research Part C*. Elsevier, 85(January), pp. 528–547. doi: 10.1016/j.trc.2017.09.025.

Li, L., Wen, D. and Yao, D. (2014) 'A Survey of Traffic Control With Vehicular Communications', *IEEE Transactions on Intelligent Transportation Systems*, 15(1), pp. 425–432. doi: 10.1109/TITS.2013.2277737.

Li, Z., Elefteriadou, L. and Ranka, S. (2014) 'Signal control optimization for automated vehicles at isolated signalized intersections', *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd, 49, pp. 1–18. doi: 10.1016/j.trc.2014.10.001.

Lillicrap, T. P. *et al.* (2015) 'Continuous control with deep reinforcement learning', *arXiv preprint arXiv:1509.02971*. Available at: http://arxiv.org/abs/1509.02971.

Liu, X., Ma, K. and Kumar, P. R. (2015) 'Towards provably safe mixed transportation systems with human-driven and automated vehicles', in *2015 54th IEEE Conference on Decision and Control (CDC)*. Osaka, Japan: IEEE, pp. 4688–4694. doi: 10.1109/CDC.2015.7402950.

de Luca, S. *et al.* (2017) 'Transportation systems with connected and non-connected vehicles: Optimal traffic control', in *2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*. IEEE, pp. 13–18. doi: 10.1109/MTITS.2017.8005660.

Luyanda, F. *et al.* (2003) 'ACS-Lite Algorithmic Architecture: Applying Adaptive Control System Technology to Closed-Loop Traffic Signal Control Systems', *Transportation Research Record: Journal of the Transportation Research Board*, 1856, pp. 175–184. doi: 10.3141/1856-19.

Makarem, L. and Gillet, D. (2011) 'Decentralized Coordination of Autonomous Vehicles at intersections', *IFAC Proceedings Volumes*. IFAC, 44(1), pp. 13046–13051. doi: 10.3182/20110828-6-IT-1002.02529.

Milakis, D., van Arem, B. and van Wee, B. (2017) 'Policy and society related implications of automated driving: A review of literature and directions for future research', *Journal of Intelligent Transportation Systems*. Taylor & Francis, 21(4), pp. 324–348. doi: 10.1080/15472450.2017.1291351.

Mladenovic *et al.* (2016) 'Intersecting our mobilities: path dependence from manually-operated semaphore to self-driving vehicles?', in *2016 IEEE International Symposium on Technology and Society (ISTAS)*. IEEE, pp. 1–6. doi: 10.1109/ISTAS.2016.7764042.

Mladenovic, M. N., Abbas, M. and McPherson, T. (2014) 'Development of socially sustainable traffic-control principles for self-driving vehicles: The ethics of anthropocentric design', in *2014 IEEE International Symposium on Ethics in Science, Technology and Engineering*. IEEE, pp. 1–8. doi: 10.1109/ETHICS.2014.6893448.

Mnih, V. *et al.* (2013) 'Playing Atari with Deep Reinforcement Learning', *arXiv preprint arXiv:1312.5602*. Available at: http://arxiv.org/abs/1312.5602.

Mousavi, S. S., Schukat, M. and Howley, E. (2017) 'Traffic Light Control Using Deep Policy-Gradient and Value-Function Based Reinforcement Learning', *arXiv preprint*. Available at: http://arxiv.org/abs/1704.08883

(Accessed: 27 January 2021).

Naumann, R., Rasche, R. and Tacken, J. (1998) 'Managing autonomous vehicles at intersections', *IEEE Intelligent Systems*, 13(3), pp. 82–86. doi: 10.1109/5254.683216.

Neuendorf, N. and Bruns, T. (2004) 'The vehicle platoon controller in the decentralised, autonomous intersection management of vehicles', in *Proceedings of the IEEE International Conference on Mechatronics, 2004. ICM '04.* IEEE, pp. 375–380. doi: 10.1109/ICMECH.2004.1364468.

Nian, R., Liu, J. and Huang, B. (2020) 'A review On reinforcement learning: Introduction and applications in industrial process control', *Computers and Chemical Engineering*. Elsevier Ltd, p. 106886. doi: 10.1016/j.compchemeng.2020.106886.

Papageorgiou, M. *et al.* (2003) 'Review of road traffic control strategies', *Proceedings of the IEEE*, 91(12), pp. 2043–2067. doi: 10.1109/JPROC.2003.819610.

Parra, I. *et al.* (2017) 'Analysis of ITS-G5A V2X communications performance in autonomous cooperative driving experiments', in *2017 IEEE Intelligent Vehicles Symposium (IV)*. Redondo Beach, CA, USA: IEEE, pp. 1899–1903. doi: 10.1109/IVS.2017.7995982.

Peirce, J. R. and Webb, P. J. (1990) 'MOVA Control of Isolated Traffic Signals - Recent Experience', in *Third International Conference on Road Traffic Control*. London, UK: IET, pp. 110–113. Available at: http://ieeexplore.ieee.org/document/114389/ (Accessed: 27 January 2021).

Pendleton, S. D. *et al.* (2017) 'Perception, planning, control, and coordination for autonomous vehicles', *Machines*, 5(1), pp. 1–54. doi: 10.3390/machines5010006.

Penic, M. A. and Upchurch, J. (1992) 'TRANSYT-7F: enhancement for fuel consumption, pollution emissions, and user costs', *Transportation Research Record*, (1360).

Perronnet, F., Abbas-Turki, A. and El Moudni, A. (2013) 'A sequenced-based protocol to manage autonomous vehicles at isolated intersections', in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. The Hague, The Netherlands: IEEE, pp. 1811–1816. doi: 10.1109/ITSC.2013.6728491.

Van Der Pol, E. and Oliehoek, F. A. (2016) 'Coordinated Deep Reinforcement Learners for Traffic Light Control', in *NIPS'16 Workshop on Learning, Inference and Control of Multi-Agent Systems*. Barcelona, Spain. Available at: http://elisevanderpol.nl/papers/vanderpol_oliehoek_nipsmalic2016.pdf (Accessed: 27 January 2021).

Porche, I. and Lafortune, S. (1999) 'Adaptive Look-ahead Optimization of Traffic Signals∗', *ITS Journal - Intelligent Transportation Systems Journal*. Taylor & Francis, 4(3–4), pp. 209–254. doi: 10.1080/10248079908903749.

PTV – Planung Transport Verkehr AG (2019) *User Manual VISSIM Version 11*. Karlsruhe, Germany. Available at: https://company.ptvgroup.com/en/ (Accessed: 27 January 2021).

Qian, X. *et al.* (2014) 'Priority-based coordination of autonomous and legacy vehicles at intersection', in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Qingdao, China: IEEE, pp. 1166–1171. doi: 10.1109/ITSC.2014.6957845.

Quinlan, M. *et al.* (2010) 'Bringing simulation to life: A mixed reality autonomous intersection', in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Taipei: IEEE, pp. 6083–6088. doi: 10.1109/IROS.2010.5651993.

Rettore, P. H. *et al.* (2019) 'Vehicular Data Space: The Data Point of View', *IEEE Communications Surveys and Tutorials*. IEEE, 21(3), pp. 2392–2418. doi: 10.1109/COMST.2019.2911906.

Rios-Torres, J. and Malikopoulos, A. A. (2016) 'A Survey on the Coordination of Connected and Automated Vehicles at Intersections and Merging at Highway On-Ramps', *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–12. doi: 10.1109/TITS.2016.2600504.

Robbins, H. and Monro, S. (1951) 'A stochastic approximation method', *The annals of mathematical statistics*. JSTOR, pp. 400–407.

Robertson, D. I. (1969) *TRANSYT: A Traffic Network Study Tool*, *Road Research Laboratory*. Crowthorne, Berkshire 1969: Road Research Laboratory. Available at: https://trl.co.uk/reports/LR253 (Accessed: 1 May

2017).

Robertson, D. I. and Bretherton, R. D. (1974) ‘Optimum control of an intersection for any known sequence of vehicle arrivals’, in *Proceedings of the 2nd IFAC/IFIP/IFORS Symposium on Traffic Control and Transportation Systems*.

Rodrigues de Campos, G. *et al.* (2017) ‘Traffic Coordination at Road Intersections: Autonomous Decision-Making Algorithms Using Model-Based Heuristics’, *IEEE Intelligent Transportation Systems Magazine*. IEEE, 9(1), pp. 8–21. doi: 10.1109/MITS.2016.2630585.

SAE International (2018) *Surface Vehicle Recommended Practice J3016*, *SAE International*. Pennsylvania, United States. doi: 10.4271/J3016_201806.

SAE International (2020) *V2X Communications Message Set Dictionary J2735_202007*. doi: 10.4271/J2735_202007.

Schepperle, H. and Böhm, K. (2008) ‘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*. IEEE, pp. 105–112. doi: 10.1109/CECandEEE.2008.88.

Schwarting, W., Alonso-Mora, J. and Rus, D. (2018) ‘Planning and Decision-Making for Autonomous Vehicles’, *Annual Review of Control, Robotics, and Autonomous Systems*, 1(1), pp. 187–210. doi: 10.1146/annurev-control-060117-105157.

Sen, S. and Head, K. L. (1997) ‘Controlled Optimization of Phases at an Intersection’, *Transportation Science*, 31(1), pp. 5–17. doi: 10.1287/trsc.31.1.5.

Shahidi, N., Au, T. and Stone, P. (2011) ‘Batch Reservations in Autonomous Intersection Management (Extended Abstract)’, in Tumer, K. et al. (eds) *AAMAS ’11 The 10th International Conference on Autonomous Agents and Multiagent Systems – Innovative Applications Track*. Taipei, Taiwan: International Foundation for Autonomous Agents and Multiagent Systems Richland, pp. 1225–1226. Available at: http://dl.acm.org/citation.cfm?id=2034498 (Accessed: 27 January 2021).

Silver, D. *et al.* (2014) ‘Deterministic policy gradient algorithms’, in.

Sims, A. G. and Dobinson, K. W. (1980) ‘The Sydney coordinated adaptive traffic (SCAT) system philosophy and benefits’, *IEEE Transactions on Vehicular Technology*, 29(2), pp. 130–137. doi: 10.1109/T-VT.1980.23833.

Steinmetz, E. *et al.* (2014) ‘Communication analysis for centralized intersection crossing coordination’, in *2014 11th International Symposium on Wireless Communications Systems (ISWCS)*. IEEE, pp. 813–818. doi: 10.1109/ISWCS.2014.6933465.

Stevanovic, J. *et al.* (2008) ‘Stochastic optimization of traffic control and transit priority settings in VISSIM’, *Transportation Research Part C: Emerging Technologies*, 16(3), pp. 332–349. doi: 10.1016/j.trc.2008.01.002.

Sutton, R. S. and Barto, A. G. (2018) *Reinforcement Learning: An Introduction*. MIT Press. Available at: https://books.google.co.uk/books?id=uWV0DwAAQBAJ.

Szepesvári, C. (2010) ‘Algorithms for Reinforcement Learning’, *Synthesis Lectures on Artificial Intelligence and Machine Learning*. Morgan & Claypool Publishers, 4(1), pp. 1–103. doi: 10.2200/S00268ED1V01Y201005AIM009.

Tachet, R. *et al.* (2016) ‘Revisiting Street Intersections Using Slot-Based Systems’, *PLOS ONE*. Edited by T. Tang, 11(3), p. e0149607. doi: 10.1371/journal.pone.0149607.

Tallapragada, P. and Cortés, J. (2015) ‘Coordinated intersection traffic management’, *IFAC-PapersOnLine*. Philadelphia: IFAC-PapersOnLine, 48(22), pp. 233–239. doi: 10.1016/j.ifacol.2015.10.336.

Thorpe, T. L. and Anderson, C. W. (1996) *Traffic Light Control Using SARSA with Three State Representations*. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.55.5406&rep=rep1&type=pdf (Accessed: 3 May 2017).

Transportation Research Board (2010) *Highway Capacity Manual 2010*. 5th Ed. Washington, D.C.:

Transportation Research Board.

Varga, N. *et al.* (2017) 'An architecture proposal for V2X communication-centric traffic light controller systems', in *2017 15th International Conference on ITS Telecommunications (ITST)*. Warsaw, Poland: IEEE, pp. 1–7. doi: 10.1109/ITST.2017.7972217.

Vasirani, M. and Ossowski, S. (2009) 'Market-based coordination for intersection control', in *Proceedings of the 2009 ACM symposium on Applied Computing - SAC '09*. New York, New York, USA: ACM Press, pp. 747–751. doi: 10.1145/1529282.1529439.

Vial, J. J. B. *et al.* (2016) 'Scheduling Autonomous Vehicle Platoons Through an Unregulated Intersection', pp. 1–14. Available at: http://arxiv.org/abs/1609.04512 (Accessed: 27 January 2021).

Wang, Y. *et al.* (2018) 'A review of the self-adaptive traffic signal control system based on future traffic environment', *Journal of Advanced Transportation*. doi: 10.1155/2018/1096123.

Watkins, C. J. C. H. (1989) *Learning from delayed rewards*. PhD dissertation, University of Cambridge. Available at: https://www.researchgate.net/profile/Christopher_Watkins2/publication/33784417_Learning_From_Delayed_Rewards/links/53fe12e10cf21edafd142e03/Learning-From-Delayed-Rewards.pdf (Accessed: 27 January 2021).

Wiedemann, R. and Reiter, U. (1992) 'Microscopic traffic simulation: the simulation system MISSION, background and actual state', *Project ICARUS (V1052) Final Report*, 2, pp. 1–53.

Wright, S. and Nocedal, J. (1999) 'Numerical optimization', *Springer Science*, 35(67–68), p. 7.

Wu, J., Abbas-Turki, A. and Moudni, A. EL (2009) 'Intersection traffic control by a novel scheduling model', in *2009 IEEE/INFORMS International Conference on Service Operations, Logistics and Informatics*. IEEE, pp. 329–334. doi: 10.1109/SOLI.2009.5203954.

Wu, W. *et al.* (2015) 'Distributed Mutual Exclusion Algorithms for Intersection Traffic Control', *IEEE Transactions on Parallel and Distributed Systems*, 26(1), pp. 65–74. doi: 10.1109/TPDS.2013.2297097.

Wu, Y. *et al.* (2016) 'Sequential game solution for lane-merging conflict between autonomous vehicles', in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. Rio de Janeiro, Brazil: IEEE, pp. 1482–1488. doi: 10.1109/ITSC.2016.7795753.

Xie, X.-F. *et al.* (2012) 'Schedule-driven intersection control', *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd, 24, pp. 168–189. doi: 10.1016/j.trc.2012.03.004.

Yan, F., Wu, J. and Dridi, M. (2014) 'A scheduling model and complexity proof for autonomous vehicle sequencing problem at isolated intersections', in *Proceedings of 2014 IEEE International Conference on Service Operations and Logistics, and Informatics*. IEEE, pp. 78–83. doi: 10.1109/SOLI.2014.6960697.

Yang, B. and Monterola, C. (2016) 'Efficient intersection control for minimally guided vehicles: A self-organised and decentralised approach', *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd, 72, pp. 283–305. doi: 10.1016/j.trc.2016.10.004.

Yang, K., Guler, S. I. and Menendez, M. (2016) 'Isolated intersection control for various levels of vehicle technology: Conventional, connected, and automated vehicles', *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd, 72, pp. 109–129. doi: 10.1016/j.trc.2016.08.009.

Yang, Z. *et al.* (2016) 'Cooperative driving model for non-signalized intersections based on reduplicate dynamic game', in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. Rio de Janeiro, Brazil: IEEE, pp. 1366–1371. doi: 10.1109/ITSC.2016.7795735.

Yau, K.-L. A. *et al.* (2017) 'A Survey on Reinforcement Learning Models and Algorithms for Traffic Signal Control', *ACM Computing Surveys*, 50(3), pp. 1–38. doi: 10.1145/3068287.

Zhao, D., Dai, Y. and Zhang, Z. (2012) 'Computational Intelligence in Urban Traffic Signal Control: A Survey', *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4), pp. 485–494. doi: 10.1109/TSMCC.2011.2161577.

Zhao, L., Malikopoulos, A. and Rios-Torres, J. (2017) 'Optimal Control of Connected and Automated Vehicles at Roundabouts: An Investigation in a Mixed-Traffic Environment', *arXiv preprint*. Available at: http://arxiv.org/abs/1710.11295.

Zhou, F., Li, X. and Ma, J. (2015) ‘Parsimonious shooting heuristic for trajectory control of connected automated traffic part I: Theoretical analysis with generalized time geography’, *arXiv:1511.04810[math.OC]*, pp. 1–40. Available at: http://arxiv.org/abs/1511.04810 (Accessed: 27 January 2021).

Zhu, M. *et al.* (2009) ‘LICP: A look-ahead intersection control policy with intelligent vehicles’, in *2009 IEEE 6th International Conference on Mobile Adhoc and Sensor Systems*. IEEE, pp. 633–638. doi: 10.1109/MOBHOC.2009.5336944.