Self-Organising Traffic Control Algorithms at Signalised Intersections

Mark David Einhorn

Dissertation presented for the degree of
Doctor of Philosophy
in the Faculty of Science at Stellenbosch University

Promoter: Prof JH van Vuuren
Co-promoter: Dr AP Burger

March 2015

Declaration

By submitting this dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: March 1, 2015

Abstract

The debilitating social, economic and environmental ramifications of traffic congestion are experienced in large cities the world over. The optimisation of traffic signal timings at signalised road intersections attempts to mitigate the extent of these adverse effects of traffic congestion by reducing the delay time experienced by vehicles in a transport network. Today, traffic signal control schemes may be classified into one of two main classes, namely *fixed-time traffic signal control strategies*, which are typically cyclic in nature, and *vehicle-actuated traffic signal control strategies*, which are typically acyclic in nature. Generally, cyclic control strategies tend to lack flexibility, and are unable to adapt to short-term fluctuations in traffic flow rates, resulting in green times that are either too long or too short. On the other hand, acyclic control strategies tend to lack coordination between intersections, resulting in vehicles being required to stop at the majority of signalised intersections they encounter.

Self-organising traffic signal control has been proposed as an attractive alternative form of control which both exhibits flexibility and facilitates a global coordination between intersections as a result of localised signal switching policies. Two examples of existing self-organising traffic signal control algorithms from the literature include an algorithm proposed by Lämmer and Helbing in 2008 and an algorithm proposed by Gershenson and Rosenblueth in 2012. These algorithms have been shown to outperform both optimised fixed-time traffic signal control techniques as well as state-of-the-art vehicle actuated traffic signal control techniques, in terms of reducing vehicle delay time in a transport network. A draw-back of both of these self-organising approaches, however, is that their effective operation relies on carefully selected parameter values; poorly selected parameter values may render these algorithms very ineffectual.

In this dissertation, three novel self-organising traffic signal traffic control algorithms are proposed. These three algorithms assume the use of existing radar detection sensors mounted at the intersection to provide the necessary input data. The radar detection sensors are capable of detecting and tracking individual vehicles approaching an intersection, providing real-time information pertaining to their physical dimensions, velocities, and ranges from the intersection in terms of both time and distance. The three traffic signal control algorithms are free of any user-specified parameters, and instead rely solely on the data provided by the radar detection sensors to inform their signal switching policies.

The first of these traffic signal control algorithms is inspired by inventory control theory, and draws parallels between the monetary costs typically considered in inventory control models and the delay time costs associated with traffic control at signalised intersections, which the algorithm attempts to minimise.

The second novel traffic control algorithm is inspired by the chemical process of osmosis in which solvent molecules move unaided from a region where they are highly concentrated, across a semi-permeable membrane, into a region of high solute molecule concentration. The algorithm models vehicles approaching an intersection as solvent molecules and the physical space available

for the vehicles to occupy once they have passed through the intersection as solute molecules. Following this analogy, the intersection is considered to be the semi-permeable membrane.

The third traffic control algorithm is a hybrid of the inventory and osmosis-inspired algorithms together with an intersection utilisation maximisation technique, which prevents unnecessary or prolonged underutilisation of an intersection.

The three novel traffic control algorithms, together with the algorithms of Lämmer and Helbing, and of Gershenson and Rosenblueth, as well as a fixed-time control algorithm, are implemented in a purpose-built microscopic traffic simulation modelling framework. Several measures are employed to evaluate the relative performances of the algorithms. These measures include the usual mean and maximum resulting delay times incurred by vehicles and the saturation level of the roadways in the transport network, as well as three novel performance measure indicators which include the mean number of stops made by vehicles, their mean normalised delay time and the mean normalised number of stops made. The algorithms are compared in the context of a linear corridor road network topology as well as a grid road network topology under various traffic flow conditions. The overall performance of the novel hybrid traffic signal control algorithm is found to be superior for the corridor road network topology, while the performance of the osmosis-inspired algorithm is found to be superior for the grid road network topology.

Uittreksel

Die negatiewe sosiale, ekonomiese en omgewingsimpak van verkeersopeenhoping word in groot stede regoor die wêreld ervaar. Die doel met die optimering van verkeersligwerkverrigting by straatkruisings is om die omvang van hierdie negatiewe impak teë te werk deur die vertraging van voertuie in 'n vervoernetwerk te verminder. Hedendaagse verkeersbeheeralgoritmes kom in een van twee hoofklasse voor, naamlik *vaste-tyd beheerstrategieë*, wat gewoonlik siklies van aard is, en *beheerstrategieë gebaseer op voertuigopsporing*, wat tipies asiklies van aard is. Oor die algemeen beskik sikliese beheerstrategieë nie oor genoegsame buigsaambeid om aan te pas by kort-termyn fluktuasies in verkeersvloei nie, wat tipies daartoe lei dat hul groentye spesifiseer wat òf te lank òf te kort is. Aan die ander kant is asikliese beheerstrategieë nie daartoe in staat om koördinasie tussen naasliggende straatkruisings te bewerkstellig nie, wat weer daartoe lei dat voertuie genoodsaak word om by die oorgrote meerderheid straatkruisings op hul pad te stop.

Die self-organiserende beheer van verkeersligte is as 'n aantrektlike, buigsame alternatief voorgestel wat in staat is om globale koördinasie tussen naasliggende straatkruisings as gevolg van gelokaliseerde seinstrategieë te bewerkstellig. Twee voorbeelde van bestaande self-organiserende verkeersbeheeralgoritmes in die literatuur is die algoritmes wat in 2008 deur Lämmer and Helbing en in 2012 deur Gershenson en Rosenblueth voorgestel is. Daar is aangetoon dat hierdie algoritmes daartoe in staat is om ge-optimeerde vaste-tyd beheerstrategieë sowel as gevorderde strategieë gebaseer op voertuigopsporing uit te stof in terme van 'n vermindering van die vertraging van voertuie in 'n vervoernetwerk. 'n Nadeel van beide hierdie self-organiserende benaderings is egter dat hul doeltreffende werkverrigting berus op versigtig-gekose parameterwaardes; willekeurige parameterwaardes mag lei na hoogs ondoeltreffende werkverrigitng van die algoritmes.

Drie nuwe self-organiserende verkeersbeheeralgoritmes word in hierdie proefskrif voorgestel. Hierdie drie algoritmes maak vir hul toevoerdata staat op die beskikbaarhed van bestaande radar opsporingsensors wat by straatkruisings geïnstalleer is. Die sensors is daartoe in staat om individuele voertuie wat 'n straatkruising nader, op te spoor, te volg en intydse data oor hul fisiese dimensies, snelhede, en afstande na die kruising (in terme van beide tyd en afstand) te lewer. Die drie algoritmes bevat geen gebruikers-gespesifiseerde parameters nie, en maak in plaas daarvan slegs gebruik van die sensortoevoerdata om hul beheerstrategieë te bepaal.

Die eerste van hierdie verkeersbeheeralgoritmes is deur die teorie van voorraadbeheer geïnspireer en maak gebruik van parallelle tussen die monetêre kostes wat tipies in voorraadbeheermodelle voorkom en die kostes in terme van vertragingstyd wat met verkeersbeheer by straatkruisings aangegaan word, en wat deur die algoritme geminimeer word.

Die tweede verkeersbeheeralgoritme is deur die chemiese proses van osmose geïnspireer, waar molekules van 'n oplossingsmiddel sonder eksterne hulp vanaf 'n gebied waar hul in hoë konsentrasie voorkom, deur 'n gedeeltelik-deurlaatbare membraan beweeg na 'n gebied waarin hul ook in hoë konsentrasie, maar in opgeloste vorm voorkom. Die algoritme modelleer voertuie

wat ’n straatkruising nader as die molekules van die oplossingsmiddel en die fisiese ruimte wat aan die ander kant van die kruising beskikbaar is om deur voertuie beset te word, as molekules in opgeloste vorm. In hierdie analogie word die kruising self as die gedeeltelik-deurlaatbare membraan beskou.

Die derde algoritme is ’n hibriede strategie waarin elemente van die eerste twee algoritmes in samewerking met ’n tegniek vir die maksimering van straatkruisingsbenutting gekombineer word, en wat wat ten doel het om onnodige of verlengte onderbenutting van die kruising te vermy.

Hierdie drie nuwe verkeersbeheeralgoritmes word, tesame met die bestaande algoritmes van Lämmer en Helbing, en van Gershenson en Rosenblueth, asook ’n vaste-tyd beheeralgoritme, in ’n mikroskopiese verkeersimulasiemodelleringsraamwerk wat spesifiek vir die doel ontwerp is, geïmplementeer. Verskeie maatstawwe word ingespan om die relatiewe werkverrigting van die algoritmes te evalueer. Hierdie maatstawwe sluit in die gebruiklike gemiddelde en maksimum vertragingstye van voertuie en die versadigingsvlak van strate in die vervoernetwerk, sowel as drie nuwe maatstawwe, naamlik die gemiddelde aantal stoppe deur voertuie, hul genormaliseerde vertragingstye en die gemiddelde, genormaliseerde aantal stoppe. Die algoritmes word in die kontekste van ’n lineêre topologie van opeenvolgende straatkruisings en ’n netwerktopologie van reghoekige straatblokke onder verskeie verkeersdigthede met mekaar vergelyk. Daar word bevind dat die nuwe hibriede algoritme die beste vaar in die lineêre topologie, terwyl die osmose-geïnspireerde algoritme die ander algortmes uitstof in die straatblok-netwerktopologie.

Acknowledgements

The author wishes to acknowledge a number of people and institutions for their various contributions towards the completion of this work.

- I wish to thank my co-promoter, Dr Alewyn Burger, for all of his programming support and inspired ideas with respect to the topic.
- I would like to thank the Departments of Logistics and of Industrial Engineering for the use of their excellent computing facilities over the past three years. I would also like to thank the staff of both departments for making the past three years some of the most enjoyable I have spent at Stellenbosch University.
- The National Research Foundation is hereby acknowledged with gratitude for funding in the form of the Scarce Skills Scholarship for doctoral students.
- To my fellow operations research post-graduate students, I say thank you. Your hard work over the past three years has proved to be both inspirational and motivational. I am proud to call you both my colleagues and my friends. I am confident that this will not be the last we see of one another and I wish you all the very best for what lies ahead.
- To those closest to me; my mother, Claire, my father, Peter, my sister, Paula, and my girlfriend, Anri. I cannot thank you enough for your unwavering support, patience, and encouragement, particularly during the more challenging times of the past three years. None of my achievements would have been possible were it not for the four of you, so thank you. I love you all.
- Finally, and by no means least, I wish to extend my deepest gratitude to my promoter, my mentor, and my friend, Professor Jan van Vuuren. This thanks extends past the three years we have worked together on this dissertation, to my very first day at Stellenbosch University, nine years ago. Professor van Vuuren unearthed an unwavering passion within me for mathematics, and in particular, for operations research. Over the past nine years, Professor van Vuuren's continued belief in my abilities, even when it was lacking on my part, has been invaluable. Apart from his contributions towards the work contained in this dissertation, his teachings on art, culture, perseverance and history have shaped me into the person I am proud to be today, and for that I am grateful. Professor van Vuuren has always told me that he is easily pleased with the very best, and I can only hope that the work contained in this dissertation does, in fact, please him. Thank you, Prof.

Table of Contents

List of Reserved Symbols

Symbol	Meaning
Symbols pertaining to microscopic traffic flow variables.	
$v_i(t)$	The speed of vehicle i at time t.
ℓ_i	The actual length of vehicle i from front bumper to rear bumper.
x_i	The longitudinal position of vehicle i.
a_i	The acceleration of vehicle i.
h_{t_i}	The time headway of vehicle i.
t_{t_i}	The amount by which the time headway of vehicle i exceeds the saturation time headway.
t_{g_i}	The time gap between the vehicle i and vehicle $i-1$ in front of it.
t_{o_i}	The occupancy time of vehicle i
h_{s_i}	The space headway of vehicle i.
x_{s_i}	The space gap of vehicle i.
$\hat{\ell}_i$	The effective length of vehicle i. This is the actual length of vehicle i together with a minimum safety gap that has to be maintained between stationary vehicles.
$\mu_i(t)$	The stopping point of vehicle i at time t.
$\epsilon_i(t)$	The position of vehicle i in the predicted queue along lane j at time t.
$d_{i,\rho_j(t)}(t)$	The distance from vehicle i to the queue position along approach lane j at time t.
$d_{i,\mu_i(t)}(t)$	The distance from vehicle i to its stopping point along approach lane j at time t.
$\phi_i(t)$	The expected delay of vehicle i at time t.
Symbols pertaining to macroscopic traffic flow variables.	
$\bar{l}$	The average length of a group of vehicles.
$\bar{h}_t$	The average time headway of a group of vehicles.
$\bar{h}_{t_{\mathrm{sat}}}$	The saturation headway of a group of vehicles departing from an intersection.
$\bar{h}_s$	The average space headway of a group of vehicles.
$\bar{v}_t$	The time-mean speed of a group of vehicles.
$\bar{v}_s$	The space-mean speed of a group of vehicles.
$\bar{v}_c$	The capacity-flow speed of a group of vehicles.
$\bar{v}_0$	The desired free-flow speed of a group of vehicles.
Q_j	The macroscopic rate of vehicle flow of road lane j.
$Q_j^{\max}$	The maximum flow rate possible along road lane j.
K_j	The macroscopic traffic density along road lane j.
K_j^{crit}	The critical traffic density along road lane j.

K_j^{jam}	The jam density along road lane j.
O_j	The macroscopic traffic occupancy along road lane j.
Symbols pertaining to intersections and intersection approach lanes.	
I_x	The displacement of intersection I from the centre of a road network along the x-axis.
I_y	The displacement of intersection I from the centre of a road network along the y-axis.
b_I	The offset value of intersection I.
F_c	The critical flow rate to capacity ratio of an intersection.
C	The duration of a fixed-time traffic signal cycle implemented at an intersection.
F_j	The flow rate to capacity ratio of approach lane j.
c_j	The capacity of approach lane j.
g_j	The green time received by approach lane j.
τ_j	The set up time implemented before approach lane j receives service.
φ_I	A count of vehicles waiting behind or approaching a red signal within a specified distance d of intersection I
ς_I	A threshold value for the number of vehicles waiting behind or approaching a red signal within a specified distance d of intersection I
u_I	The minimum allowable green time implemented at intersection I.
$Q_j^{\text{arr}}(t)$	The average vehicle arrival rate along approach j at time t.
$Q_j^{\text{dep}}(t)$	The average vehicle departure rate along approach j at time t.
$o_j(t)$	The priority index assigned to approach lane j at time t.
$\hat{n}_j(t)$	The anticipated number of vehicles expected to be served at the maximum flow rate along approach lane j at time t.
$\hat{g}_j(t)$	The anticipated green time required to clear all currently queued vehicles along approach lane j as well as those which will join the queue during the remaining set-up phase and while the queue is being cleared.
$\sigma(t)$	The index of the approach lane currently receiving service at time t.
τ_σ	The penalty term associated with terminating service to approach lane σ.
n_j^{crit}	The critical anticipated queue length along approach lane j.
$z_j(t)$	The service interval for approach j at time t which corresponds to the time interval between the end of the last green time received by approach j and the start of the next service period it will receive.
$\mathfrak{I}(t)$	The set of all detected vehicles along all approach lanes at time t.
$\mathfrak{C}_j(t)$	The set of all vehicles present on approach lane j at time t.
$\mathfrak{Q}_j(t)$	The set of all vehicles present on approach lane j at time t which are either queued or are predicted to become queued.
α_j	The length, in metres, of approach lane j.
$\rho_j(t)$	The queue position of predicted queued vehicles along approach lane j at time t.
$\beta_j(t)$	The number of vehicles predicted to become queued along approach lane j at time t (*i.e.* the magnitude of $\mathfrak{Q}_j(t)$).
$\gamma_j(t)$	The green time to clear all predicted queued vehicles along approach lane j at time t.
$\kappa_j(t)$	A binary variable indicating whether or not approach lane j is currently receiving service at time t.
$\delta_j(t)$	The demand of approach lane j at time t, where $\delta_j(t) = \sum_{i \in \mathfrak{C}_j(t)} \hat{\ell}_i$.

$\omega_j(t)$	The availability associated with approach lane j at time t, where $\omega_j(t) = \alpha_{j'} - \delta_{j'}(t)$. Here, j' is the probable exit lane associated with approach lane j.
$\pi_j(t)$	The pressure exerted on an intersection by approach lane j at time t, where $\pi_j(t) = \delta_j(t) + \omega_j(t)$.
$\theta_j(t)$	The throughput of approach lane j at time t.
η_j	The assumed rate of departure of stationary vehicles from a queue along approach lane j.
Symbols pertaining to individual service phases of a traffic signal cycle.	
$\mathfrak{P}$	The set of all service phases of a traffic signal cycle implemented at an intersection.
$\mathfrak{A}_m$	The set of all approach lanes served during phase m of a traffic signal cycle.
E_m	The set of all probable exit lanes utilised during phase m of a traffic signal cycle.
$\Gamma_m(t)$	The required green time of phase m of a traffic signal cycle at time t, where $\Gamma_m(t) = \max_{j \in \mathfrak{A}_m} \gamma_j(t)$.
$\chi_m(t)$	The remaining green time of phase m of a traffic signal cycle at time t.
$\tau_m(t)$	The remaining setup time of phase m of a traffic signal cycle at time t.
$\Phi_m(t)$	The expected total delay of implementing the required green time of phase m at time t, where $\Phi_m(t) = \sum_{i \in I(t)} \phi_i(t)$
$\Pi_m(t)$	The pressure exerted on an intersection by phase m of a traffic signal cycle at time t, where $\Pi_m(t) = \sum_{j \in A_m} \pi_j(t)$.
$\Delta_m(t)$	The combined demand of all approach lanes served during phase m of a traffic signal cycle at the start of service to the phase at time t^*, where $\Delta_m(t) = \sum_{j \in \mathfrak{A}_m} \delta_j(t^*)$.
$\Omega_m(t)$	The combined availability associated with all approach lanes served during phase m at the start of service to phase m at time t^*, where $\Omega_m(t) = \sum_{j \in A_m} \omega_j(t^*)$.
$\Theta_m(t)$	The combined throughput of all approach lanes served during phase m of a traffic signal cycle at time t, where $\Theta_m(t) = \sum_{j \in \mathfrak{A}_m} \theta_j(t)$.

List of Acronyms

ALLONS-D: Adaptive Limited Look-ahead Optimisation of Network Signals

I-TSCA: Inventory Traffic Signal Control Algorithm

OPAC: Optimised Policies for Adaptive Control

O-TSCA: Osmosis Traffic Signal Control Algorithm

SATURN: The Simulation and Assignment of Traffic in Urban Road Networks

SCATS: Sydney Coordinated Adaptive Traffic System

SCOOT: Split Cycle Offset Optimisation Technique

TRANSYT: Traffic Network Study Tool

UTOPIA: Urban Traffic Optimisation by Integrated Automation

VISSIM: (a German acronym for) Traffic In Towns: Simulation

List of Figures

List of Tables

List of Algorithms

CHAPTER 1

Introduction

Contents

1.1 Background

Traffic congestion is a phenomenon experienced in most cities around the world and can have debilitating economic, environmental, and social ramifications, depending on the severity of the congestion. The main cause of congestion may be attributed to the volume of traffic being very close to, or exceeding, the maximum capacity of a road or entire road network [40]. The direct negative monetary implications of traffic congestion are felt the world over, and are largely due to man-hours lost by the working force as well as the additional fuel burned while vehicles are idle in congested traffic conditions. In a 2012 survey conducted in the United Kingdom by the Centre for Economics and Business Research and the traffic information company Inrix, it was estimated that traffic congestion costs the UK economy £4.3 billion per year [92]. Of this total, £426 million was attributed to "wasted" fuel, while the cost in terms of lost time was estimated at £2.7 billion for commuters (£331 per commuter per year), and £1.1 billion for business or freight vehicles. Traffic congestion can, however, have far greater reaching effects on economic growth as highlighted by Matthias Sweet [91], a researcher at the McMaster Institute for Transportation and Logistics at McMaster University. He explains that traffic congestion may result in staff requiring higher wages to compensate for the time spent in adverse traffic conditions, and in some cases traffic congestion may even lead to people searching for new jobs, which will require them to spend less time in traffic. This makes it difficult to match the right workers to the best jobs, which can lead to economic inefficiencies. Environmentally, it is commonly known that traffic congestion increases fuel consumption and therefore CO_2 emissions [7]. It has been found that travelling at a steady-state velocity will yield much lower emissions and fuel consumption when compared to the stop-and-go driving. Therefore, by reducing the stop-and-go driving patterns associated with congested traffic, CO_2 emissions may be reduced considerably.

Numerous strategies have been proposed in recent years for mitigating the debilitating effects of traffic congestion. One such approach, which is especially applicable to inner city commuting,

is the attempted optimisation of traffic signal timings at signalised intersections. Improved and efficient signal timings have the ability to reduce driver delay times by effectively utilising intersection capacity and allowing for the formation and propagation of "green waves" (platoons of vehicles travelling unimpeded through several adjacent intersections displaying green signals). This reduces the stop-and-go driving patterns associated with congested traffic which drivers in Los Angeles, Mexico City, India, China, Singapore, and Johannesburg listed as their most serious commuter pain in the IBM 2011 Global Commuter Pain Survey [42].

Signalised traffic control and its attempted optimisation within a traffic network have been the focus of many a study across several different scientific disciplines, including engineering, operations research, physics and statistics. Today, two distinct types of predominant traffic signal control exist: *fixed-time control* and *vehicle actuated control.* Fixed-time control was the earlier of the two approaches. It involves the optimisation of several traffic signal cycle parameters, such as the duration of the cycle itself, the duration of the various green times which comprise the cycle, and the offset of green times at adjacent intersections, in an attempt to facilitate coordination in a traffic network [33, 60, 68, 99]. These parameters are typically optimised off-line for assumed average traffic flows, such as morning and afternoon rush hours [53, 54]. A disadvantage of this approach, however, is that traffic signal timings are set for assumed mean traffic demands which are rarely actually met and as a result are typically too rigid to respond to sudden fluctuations in vehicle demand away from an assumed mean. The traffic signal timings are therefore often too long or too short, resulting in an inefficient utilisation of intersection capacity and thus avoidable vehicle delays.

Vehicle actuated control, on the other hand, seeks to adapt to variations in the average traffic demand over a given time horizon by employing some form of vehicle detection mechanism to provide input to the traffic signal control algorithm [54]. These data are then used to determine when to switch between signal phases. Two prominent examples of such control techniques are the *Split Cycle Offset Optimisation Technique* (SCOOT) [50] and the *Sydney Coordinated Adaptive Traffic System* (SCATS) [62]. While these vehicle actuated control techniques are able to perform on-line or real-time optimisation operations, they remain largely centralised, attempting to determine optimal cycle lengths, green time splits and cycle offsets of adjacent intersections based on prevailing traffic conditions as interpreted by upstream vehicle detectors. A disadvantage of this approach toward traffic signal control is that the problem of optimal control of switched network flows is known to be NP-hard [79].

Self-organising traffic signal control has been proposed as an attractive alternative to overcoming the disadvantages mentioned above as it leads to the *emergence* of favourable coordination among signalised intersections within a traffic network. De Wolf and Holvoet [22] have provided the following working definition of self-organisation based upon the historical use of the concept within the literature:

> Self-organisation is a dynamical and adaptive process where systems acquire and maintain structure themselves, without external control.

The 'structure' mentioned above may be spatial, temporal or functional, while 'no external control' refers to the absence of direction, manipulation, interference, pressures or involvement from outside the system [22]. This, however, does not preclude data inputs from outside the system as long as these inputs are not instructions. Through a comprehensive literature study, De Wolf and Holvoet [22] identified four characteristics considered to be of central importance to a self-organising system. In line with the 'organisation' concept of self-organisation, the first characteristic is that the system should *facilitate an increase in order* from semi-organised or completely random initial conditions [70]. Organisation is described in [88] as the arrangement of

selected parts so as to promote a specific function. In essence, organisation may then be viewed as an increase in the order of the system behaviour which enables the system to acquire a spatial, temporal, or functional structure. The second important characteristic of self-organisation is *autonomy*, or, more specifically, the absence of external control [70, 88]. The third characteristic is that of *adaptability* or *robustness*. A self-organising system is expected to adapt to changes or perturbations to the input data or external conditions of the system autonomously and possess the ability to exhibit a large variety of behaviours as well as being able to make an appropriate selection from these behaviours [47]. In [70], a system is considered to be adaptable if "a change in the environment may influence the same system to generate a different task, without any change in the behavioural characteristics of its constituents." The final characteristic of a self-organising system is that it must be *dynamic*. This dynamism requires a self-organising system to be *far-from-equilibrium* [22]. A far-from-equilibrium system is relatively more fragile and sensitive to changes in its immediate environment but also more dynamic and capable of adaptation to counter these changes and maintain the desired system structure.

As well as providing a working definition for self-organisation, the following working definition for emergence was also provided in [22], again based on the historic use of the concept in the relevant literature:

> A system exhibits emergence when there are coherent emergents at the macro-level that dynamically arise from interactions between the parts at the mico-level. Such emergents are novel with respect to the individual parts of the system.

Again, through a comprehensive literature study, the most important characteristics of a system capable of exhibiting emergence were identified in [22]. It was stated that the most important of these characteristics is the so-called *micro-macro effect* [22, 47, 88]. The micro-macro effect refers to the system-wide emergents (*i.e.* properties, behaviours, structures or patterns) which are observed at a macro-level of abstraction as a result of the actions and interactions of and between the individual entities of the system at a lower, micro-level. The second defining characteristic of an emergence system is *radical novelty*. This means that the individual entities at the micro-level admit no explicit representation of the emergent global behaviour. In non-reductionism terms this means that the resulting global behaviour of the system may not be directly described by the behaviour of its individual parts [20] — *i.e.* the whole is greater than the sum of its parts [75]. The third characteristic of an emergence system is that of *coherence* (or organisational closure [47]) which refers to a correlation of the lower level individual entities or components of the system into a higher level unity, *i.e.* correlations between components are required to reach a coherent whole [8]. A further characteristic of emergence systems is that they possess *interacting parts*. Without interacting parts, emergent macro-level behaviour will not arise [47]. Like a self-organising system, an emergence system is dynamic in nature in the sense that various emergent behaviours may arise at certain points in time as the system changes and evolves [22]. The sixth characteristic of an emergence system is that it exhibits *decentralised control* [47, 75]. Decentralised control implies that there is no central form of control over the system — *i.e.* no single part of the system dictates or directs the macro-level behaviour of the system. Instead, only local, lower-level interactions and mechanisms of the constituent parts of the system (which are themselves controllable) are responsible for the emergent global behaviour of the system. This characteristic of decentralised control is a direct consequence of the radical novelty which is required for emergence. A seventh characteristic exhibited by all emergence systems is that of a *bidirectional link* between the micro- and macro-level of the system [75, 88]. While the parts of the micro-level give rise to macro-level behaviour (as described by the micro-macro effect characteristic) so the emergent macro-level behaviour may influence the micro-level parts. The eighth and final characteristic of emergence systems is that they must

exhibit *robustness* and *flexibility* [47, 75]. The fact that decentralised control and radical novelty are prerequisites for an emergence system implies that a single entity cannot be a single point of failure for the system as a whole, *i.e.* the failure or replacement of a single entity will not result in the complete failure or collapse of the emergent behaviour, implying that emergents are robust and insensitive to perturbations or failures. An increase in the number of single entity failures may result in a decrease of system performance, but this decrease is typically gradual, without a sudden loss of complete function due to the flexibility of the system.

It should be noted that self-organisation does not always lead to emergence and that emergence does not always require self-organisation. It has, however, been been stated in the literature that self-organising traffic signal control can lead to emergence. In self-organising traffic signal control, traffic signal timings of individual intersections are governed by a predetermined set of algorithmic rules which are free of any external influences or control, thus satisfying the 'autonomy' requirement of a self-organising system and the 'decentralised control' requirement of an emergence system. In the paradigm of self-organising traffic signal control there is no communication between adjacent intersections and no explicit attempt is made to achieve coordination among intersections. Instead, each set of traffic signals simply adjusts its signal timings according to the interactions between itself and the vehicles requiring service within its local control domain, thus satisfying the 'adaptability' and 'dynamism' requirements of a self-organising system as well as the 'dynamism' and 'robustness' requirements of an emergence system. As a result of the interaction between the vehicles and signals in a traffic network, there is, however, an implicit interaction between adjacent traffic signals. This is because vehicles departing from the local control domain of a set of traffic signals enter the control domain of an adjacent set of traffic signals, thus satisfying the 'interacting parts' requirement of an emergence system. As a result of effective vehicle detection and local traffic signal switching operations, a natural, global coordination may emerge among sets of traffic signals, facilitating the formation and propagation of *green waves* through the traffic network. This satisfies the 'increase in order' requirement of a self-organising system together with the 'micro-macro effect', 'coherence' and 'bidirectional link' requirements of an emergence system.

Self-organising traffic signal control strategies have been shown to outperform both optimised fixed time control strategies [27, 37, 38, 53, 54, 103] and state-of-the-art centralised traffic responsive systems [54] in terms of minimising vehicle delay. Two effective self-organising traffic signal control algorithms from the literature are those of Lämmer and Helbing [53, 54] and Gershenson and Rosenblueth [38]. The self-organising traffic signal control approach by Lämmer and Helbing [53, 54] employs an optimisation strategy which seeks to serve alternate intersection approaches as quickly as possible (based on approach priority values), as well as a stabilisation strategy which ensures that queues along intersection approaches do not grow exceedingly long before receiving service. The self-organising traffic signal control approach of Gershenson [37], Gershenson and Rosenblueth [38] and Zubillaga et al. [103] also serves intersection approaches in a priority-based manner, with platoons of vehicles receiving a higher priority in an attempt at facilitating the formation and propagation of *green-waves*.

1.2 Informal problem description

While both self-organising approaches described above can be very effective at minimising vehicle delay under certain prevailing traffic conditions, they rely on several user-defined parameters to ensure their effective operation. Poorly selected parameter values can render the self-organising traffic control algorithms ineffectual, as was demonstrated in [28]. Both self-organising traffic control approaches assume the use of some vehicle detection mechanism, but they do not categor-

ically specify what form of detection is used, nor do they state the capabilities or short-comings of the detection equipment. Finally, both self-organising traffic control approaches are based on a number of simplifying assumptions such as that no vehicle acceleration takes place (they either move at a constant speed or are stationary), that all vehicles travel at the same speed and that vehicles are assumed to be of uniform size. These assumptions are made due to the fact that the self-organising traffic control algorithms rely more on the presence of vehicles along an intersection approach rather than on their individual characteristics, such as their speeds, sizes and distances from the intersection.

Effective and accurate vehicle detection is central to effective and efficient vehicle actuated traffic signal control. Since its introduction during the early 1960s the most common type of vehicle detection sensor used in vehicle actuated control to inform signal switching policies has been the inductive loop detector [51]. The inductive loop detector consists of a wire sensor loop which is embedded in the road pavement upstream from the intersection. Vehicles passing over or stopped within the detection zone of an inductive loop detector cause a disturbance within the magnetic field of the sensory loop by decreasing its inductance [51]. If the magnitude of this decrease in inductance is above a certain predetermined threshold, it is detected by the loop detector unit which is responsible for monitoring and energising the loop [27]. This loop detector then sends an output signal to the controller unit which is responsible for the implementation of the logic which determines the switching of the traffic signals [77].

While inductive loop detectors are by far the most commonly used form of vehicle detection today, they are by no means the only available option. Examples of more technically advanced alternatives include video image processing, microwave radar, infrared sensors, ultrasonic sensors and passive acoustic array sensors [69]. In this dissertation, the vehicle detection technique assumed is that of radar detection. In particular, the *SmartSensor Advance Extended Range* [97] radar detection unit (shown in Figure 1.1) manufactured by *Wavetronix* [98], or a unit similar to it in terms of capability, is assumed to provide all relevant vehicle detection data.

FIGURE 1.1: *The* SmartSensor Advance Extended Range *[97] radar detection unit.*

When mounted at an intersection, as shown in Figure 1.2, the radar detection unit provides a detection range of approximately 275 metres upstream from the intersection across multiple lanes and is capable of tracking the speed, range and time of arrival at the intersection of each vehicle it detects [98] (see Figure 1.3).

The radar detection equipment is therefore capable of detecting changes in speeds of vehicles as well as vehicle lane changes and is able to relay these live, dynamic data to the relevant traffic signal controllers accordingly. In addition to providing information pertaining to individual

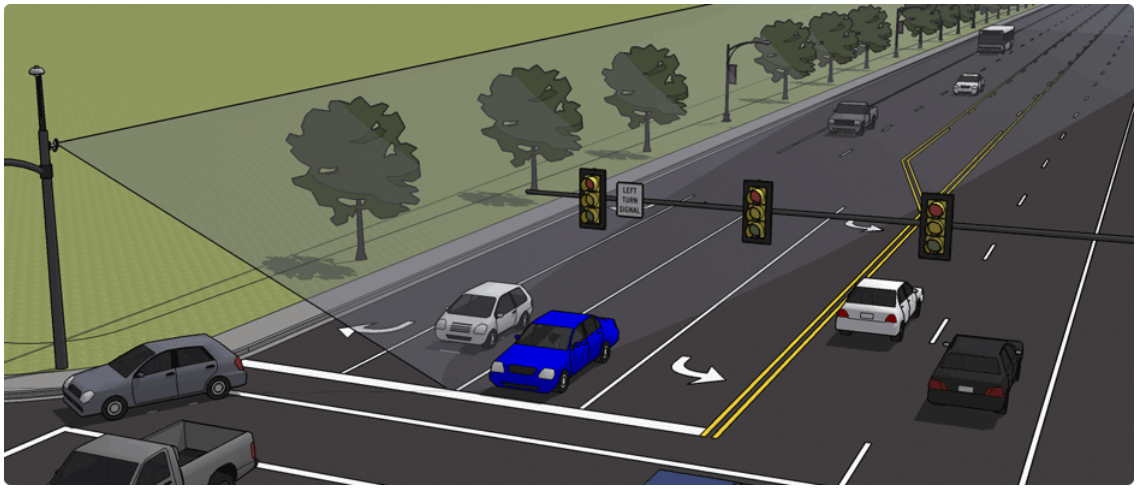


FIGURE 1.2: *A radar mounted at an intersection and its associated detection zone [98].*

vehicle speeds, ranges and estimated arrival times, the radar detection equipment can also detect and provide information on the physical dimensions of individual vehicles as well as the space between vehicles and vehicle platoons. This stands in stark contrast to inductive loop detectors which are only capable of providing information about vehicle presence along a road, and in some scenarios, instantaneous vehicle speed.

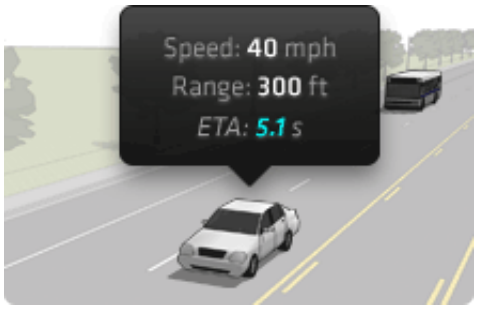


FIGURE 1.3: *An example of the vehicle-specific data provided by the radar detection equipment [98].*

Another advantage of a mounted radar detector over an inductive loop detector is that it is a *non-intrusive* form of detection. Inductive loop detectors (and other intrusive forms of detection) are installed directly in the road pavement surface and require saw-cuts or core-drilled holes that weaken the road surface [69, 98]. Their installation requires roads to be closed, disrupting traffic flow. Furthermore, in the case that the device fails or the roadway requires resurfacing, it is necessary to once again disrupt traffic to repair and reinstall the device. Radar detection devices, on the other hand, are mounted above the road surface, making them easy to install and maintain. Furthermore, they may be replaced quickly and easily, or reconfigured to accommodate roadway changes [98]. An advantage that radar detectors have over other non-intrusive means of detection (*e.g.* video image processing) is that they are more robust to inclement weather and light conditions. This is because the length of the radar's electromagnetic wave is much longer than the wavelength of light, allowing it to propagate through rain, snow, fog and dust without becoming distorted [98].

The problem considered in this dissertation may be described as attempting to find an answer to the following research question and to motivate the answer scientifically: Is it possible for a self-organising traffic signal control algorithm (which makes use of data provided by radar

detectors as mentioned above) to be free of any user-defined parameters and still result in the emergence of effective coordination among intersections in a traffic network, and if so, is such a self-organising traffic signal control algorithm capable of more effective reductions of vehicle delay time than other algorithms proposed in the literature?

1.3 Scope and objectives

In order to develop novel self-organising traffic signal control algorithms and compare their effectiveness to to those of various traffic signal control algorithms in the literature, in terms of their propensity to minimise vehicle delay time as far as possible and facilitate coordination among intersections so as to reduce the number of stops made by vehicles in the system, the following objectives are pursued in this dissertation:

1. To *perform* a comprehensive study of the literature pertaining to the various fields considered in this study, including reviews of
   (a) fundamental traffic flow theory and the dynamics of vehicle movement and delay at signalised intersections,
   (b) previously proposed methods of fixed and vehicle-actuated methods of traffic signal control,
   (c) self-organisation and previously proposed self-organising traffic signal control algorithms,
   (d) computer simulation modelling approaches and techniques with a focus on microscopic traffic simulation modelling.
2. To *investigate and compare* various traffic signal control algorithms in a simulated environment for a variety of road network topologies and prevailing traffic conditions. This requires
   (a) the design and implementation of a microscopic traffic simulation modelling framework in which both pre-existing and novel traffic signal control algorithms (which assume the use of radar detection equipment as mentioned above) may be implemented,
   (b) defining a standardised set of performance measure indicators which may be used to compare the effectiveness of the various traffic signal control algorithms, and
   (c) performing statistical analyses on the simulation results in order to rank a number of popular traffic signal control algorithms across the different performance measure indicators for several road network topologies and prevailing traffic conditions at a specified level of statistical significance.
3. To *present* the findings of the study in a scientific manner together with an in-depth analysis and interpretation of the results and their consequences.

The microscopic traffic simulation model built for the purpose of this study attempts to replicate and recreate real-world traffic situations as accurately as possible. For this reason, varying characteristics specific to each vehicle which enters the traffic network are incorporated, rather than assuming fixed, constant characteristics for every vehicle as is the case in the majority of the relevant literature. These characteristics include the physical length of the vehicles, their desired speeds, destination choices and rates of acceleration. The model allows for vehicles to change lanes in preparation to turn and facilitates both permissive and exclusive right-hand turns at

intersections. It is assumed, however, that all vehicles obey the traffic signals absolutely and that no collisions or breakdowns of vehicles occur. Furthermore, the model does not account for pedestrians in the system or exclusive pedestrian signal phases.

1.4 Dissertation organisation

This dissertation comprises eight chapters. This first chapter serves to provide the reader with an introduction and background to one of the three main themes of the dissertation, namely self-organisation and emergence (the other two being traffic flow theory and computer simulation modelling). An informal problem description was provided and the scope and objectives of the study were described.

In Chapter 2, the history of the fundamentals of traffic flow theory is documented. Microscopic traffic flow theory and all associated variables and characteristics are considered first as well as car-following models and the dynamics of vehicle delay at signalised intersections. This is followed by an analogous discussion on macroscopic traffic flow theory in which the *fundamental relation of traffic flow theory* is highlighted and illustrated by means of fundamental diagrams.

Chapter 3 serves to provide the reader with a background on computer simulation modelling. The chapter opens with a general definition of computer simulation modelling and an overview of the various types of simulation modelling approaches available. This is followed by a discussion on the advantages and disadvantages associated with simulation modelling. The final part of the chapter is dedicated to traffic simulation modelling specifically, focusing on micro-, meso- and macroscopic traffic simulation modelling approaches and the various commercially available software packages which harness them.

Chapter 4 contains an in-depth description of the microscopic traffic simulation modelling framework which was designed and built for the purpose of this study. Particular attention is afforded to the steps that were taken to ensure that the data provided by the radar detection equipment described in §1.2 are accurately incorporated into the modelling framework, allowing them to be utilised by the various traffic signal control algorithms. The approach adopted with respect to modelling the characteristics of the vehicles which populate the road network is described and a motivation is provided for the inclusion of these characteristics. The chapter closes with the verification and validation processes followed to ensure that the correct model for the study had been built and that the model had been built correctly.

In Chapter 5, existing traffic signal control paradigms are considered. The chapter opens with a discussion on the various fixed-time and vehicle-actuated control techniques that prevail in the literature. This introductory section is followed by a detailed description of three existing traffic signal control techniques which are implemented in this study. The first is an optimised fixed-time control strategy proposed in [94] for equalising the degree of saturation along all intersection approaches. The second technique is the self-organising traffic signal control algorithm of Gershenson and Rosenblueth [38] mentioned earlier in §1.1. The final technique is the self-organising traffic signal control algorithm of Lämmer and Helbing [54].

Three novel self-organising traffic signal control algorithms are proposed in Chapter 6. The first algorithm is inspired by inventory theory. The chapter opens with a brief overview of inventory theory and establishes parallels between the monetary costs incurred in typical inventory control models and vehicle delay time costs experienced in signalised traffic control. This is followed by a description of the algorithm itself and how it assimilates the data provided by radar detection sensors to inform signal switching decisions based upon an inventory control methodology. The

second algorithm is inspired by the chemical process of osmosis. A discussion of the basic fundamentals of osmosis and how they translate to signalised traffic control is presented, and this is followed by a comprehensive description of the working of the algorithm. Once again it is assumed that all input data to the algorithm are provided by the aforementioned radar detection sensors. The chapter closes with a description of the third novel traffic signal control algorithm. This algorithm is a hybrid procedure which combines the inventory-inspired traffic signal control algorithm with the osmosis-inspired traffic signal control algorithm in an attempt to exploit the best features of both. In addition to combining the two aforementioned algorithms, the hybrid algorithm incorporates a supervisory mechanism in an attempt to maximise intersection usage.

The focus in Chapter 7 falls on testing the six traffic signal control algorithms (the three previously proposed algorithms and the three novel algorithms) considered in Chapters 6 and 7, respectively. The chapter opens with a detailed description of the experimental design adopted and the performance measure indicators considered. This is followed by a presentation of the simulation results obtained for the various test instances investigated and their associated analyses and interpretations.

The final chapter of this dissertation, Chapter 8, contains a brief summary of the work presented as well as an appraisal of the contributions made. Conclusions are drawn with respect to the effectiveness of the various traffic signal control algorithms tested. The chapter closes with pertinent recommendations for further work related to self-organising traffic signal control algorithms.

CHAPTER 2

Traffic Flow Theory

Contents

This chapter contains a brief history of traffic flow theory, from its inception through to present day practices. It includes discussions and analyses of various traffic flow theories which are central to the numerous traffic flow models available in the literature. The chapter opens with an introduction to traffic flow theory and its origins in §2.1. A distinction is made between the two main arms of traffic flow theory, namely *microscopic traffic flow* and *macroscopic traffic flow* and their associated variables. The characteristics of both these theories are discussed in §2.2 and §2.3, respectively, as well the dynamics of traffic flow in general. This chapter therefore serves as a primer for the traffic flow models adopted later in this dissertation.

2.1 Introduction

Traffic flow theory is believed to have originated during the early 1950s [63], and is largely attributed to the work of Wardrop [96] who described traffic flows using mathematical and statistical expressions. The field continued to evolve over the next decade and two important examples of the progress made during this period include the fluid-dynamic model of Lighthill, Whitham and Richards for traffic flows [59, 84], which has formed the cornerstone of numerous macroscopic traffic related theories and models since, and the car-following experiments and theories of the General Motors' research laboratory [16, 35, 36, 46]. Interest in the field waned over the next few decades, however, before being restored during the early 1990s. Today, the field of traffic flow theory, including basic research and applications, has greatly diversified to

incorporate a wide range of modelling influences, drawing from fields of study such as sociology, psychology, environmental studies and economics, to name but a few [63].

According to Hoogendorn and Knoop [49], traffic flow theory comprises analyses and descriptions of the fundamental characteristics of traffic flows, such as road capacities, flow and density relationships, and headway distributions. The theory also extends to include the effects of external factors, such as weather, traffic control policies and driver behaviour on the aforementioned characteristics. Traffic flow theory is divided into two main fields, namely *microscopic traffic flow theory* and *macroscopic traffic flow theory*. Microscopic traffic flow theory encompasses the flow, speed and density associated with individual vehicles along a roadway, while in macroscopic traffic flow theory one assumes a more aggregated view, considering the flow, speed and density associated with groupings or flows of numerous vehicles as units.

2.2 Microscopic traffic flow theory

Certain characteristics inherent to the vehicle itself, as well as its driver, are associated with each vehicle in a traffic flow. When a description of the flow of vehicles comprises such individual vehicle characteristics, these characteristics are called *microscopic* and the dynamics of such traffic flows are described in terms of the underlying interactions between the drivers and their vehicles with one another [63].

The participation of a vehicle in traffic flow is largely based on the behavioural aspects of its driver, and for this reason, theory and models have been developed for incorporating these human factors into microscopic descriptions of traffic flows. One such example is the theory of *psycho-spatial* models [48] which incorporates insight from perceptual psychology to show that drivers are subject to certain limits in their perception of the stimuli to which they respond [93]. This incorporation of human influencing factors, however, greatly increases the associated model complexity [63] and for this reason, many traffic flow theories rather opt to model vehicle-driver combinations as single entities, only taking into account certain vehicle-related traffic flow characteristics.

2.2.1 Microscopic traffic flow variables and characteristics

When considering individual vehicles, several variables are associated with each vehicle travelling in a traffic stream. These variables include the *length* of vehicle i, denoted by ℓ_i, the *longitudinal position*[1] of the vehicle, denoted by x_i, the *speed* of the vehicle, $v_i = \mathrm{d}x_i/\mathrm{d}t$, and its *acceleration*, $a_i = \mathrm{d}v_i/\mathrm{d}t = \mathrm{d}^2x_i/\mathrm{d}t^2$.

Microscopic speed characteristics are considered those speed characteristics of individual vehicles passing a point or short road segment during a specified time period [65]. The speed of an individual vehicle is influenced by its immediate environment and vehicles may be required to accelerate or decelerate as a result of other vehicles along a roadway, interrupted flow situations (*e.g.* stop streets, signalised intersections, *etc.*) or roadway design features. It is common practice only to consider the acceleration capabilities of a vehicle and not any other external influencing factors such as the earth's gravitational pull, road and wind friction, and centrifugal forces [63].

The individual vehicle *headways* may include *time headways* as well as *distance* or *space headways*. The time headway is considered one of the most important microscopic traffic flow characteristics as minimal time headways directly determine the capacity of a road section [49]. The

[1] The position x_i of a vehicle is typically taken to be the position of its rear bumper [63].

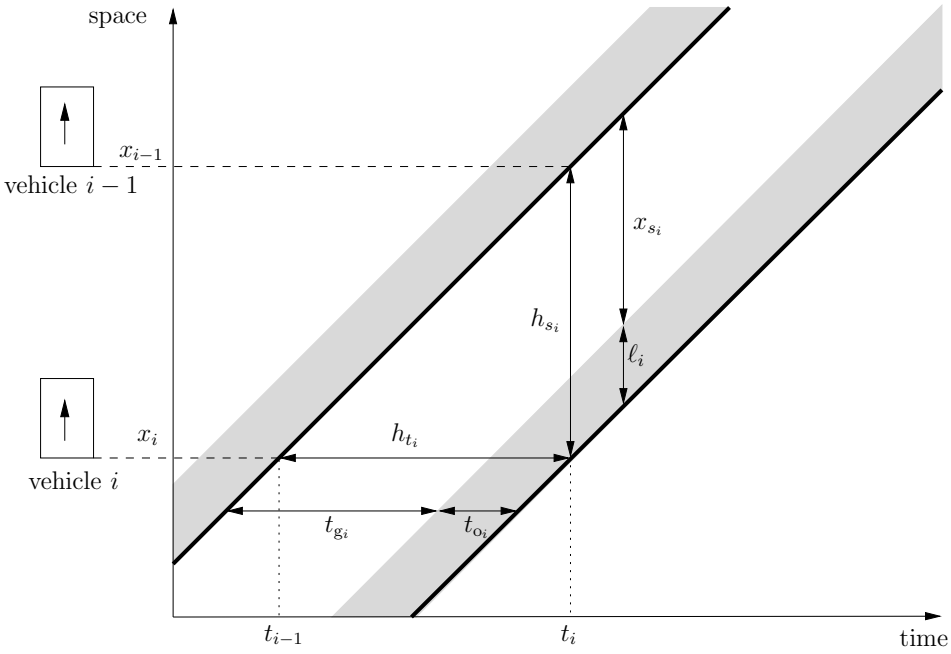


FIGURE 2.1: *A time-space diagram showing the trajectories of two vehicles (i and $i-1$) as well as the time and space headways of vehicle i (diagram based on a graphic in [61]).*

time headway h_{t_i} of vehicle i is the difference between the passage times of its rear bumper and the rear bumper of vehicle $i-1$ in front of it across a fixed point along a roadway. This time headway is expressed as the sum of a time gap t_{g_i} and an occupancy time t_{o_i}, that is $h_{t_i} = t_{g_i} + t_{o_i}$. In [63], the time gap is described as the amount of time necessary for the front bumper of vehicle i to reach the current position of the rear bumper of vehicle $i-1$ in front of it, travelling at its current speed, while the occupancy time is the time required for vehicle i to traverse its own length, *i.e.* $t_{o_i} = \ell_i / v_i$. In [49], the time gap is referred to as the *net time headway* and is considered particularly important when analysing and modelling the amount of space required by a driver to perform an overtaking manoeuvre, also known as critical gap analysis, while the sum of the time gap and occupancy time is referred to as the *gross headway*.

Analogously, a *space headway*, h_{s_i} is associated with vehicle i. This space headway is the distance between the rear bumper of vehicle i and the rear bumper of vehicle $i-1$ in front of it [49], and comprises a *space gap* x_{s_i} and its own length ℓ_i, that is $h_{s_i} = x_{s_i} + \ell_i$. Again, this space gap is sometimes referred to as the *net space headway*, while the sum of the space gap and the vehicle length is known as the *gross space headway* [49]. This space headway is considered the primary microscopic characteristic of density because of its direct relationship to time headways [65]. It is highlighted in [49] that time headways are local microscopic characteristics in the sense that they relate to the behaviour of an individual vehicle and are measured from a fixed point along a road section, whereas space headways are instantaneous in the sense that they are measured at a specific point in time. From the defining expressions for h_{t_i} and h_{s_i}, it may be seen that time and space headways are highly correlated. In particular,

$$\frac{h_{s_i}}{h_{t_i}} = \frac{x_{s_i}}{t_{g_i}} = \frac{\ell_i}{t_{o_i}} = v_i.$$

The relationship between the time and space headways may be visualised by a so-called *time-space diagram*, such as the one shown in Figure 2.1. In the figure, the positions of vehicles i and $i-1$ are plotted with respect to time, tracing out their respective *trajectories*. The speeds of the

vehicles may be found by taking the tangents of their trajectories. For simplicity, the vehicles are assumed to travel at the same constant speed in Figure 2.1, resulting in parallel trajectories.

2.2.2 Car-following models

Car-following models attempt to describe how one vehicle follows another while travelling along a road section and incorporate the three aforementioned microscopic traffic flow characteristics. Assuming the notation presented above, a general car-following situation is depicted in Figure 2.2.

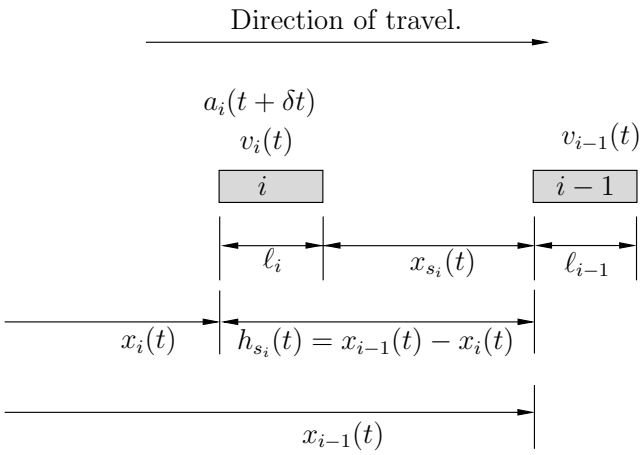


FIGURE 2.2: *Car-following theory notations and definitions.*

In Figure 2.2 the rate of acceleration (a_i) of the following vehicle (vehicle i) occurs at time $t+\delta t$ and not at time t. Here, δt represents the time interval between a car-following situation at time t and the point in time when the driver of vehicle i decides to accelerate or decelerate at time $t+\delta t$ in response to this situation. This time interval is often referred to as the *reaction time* of the driver [65]. The relative velocity of the lead vehicle with respect to the following vehicle is denoted as $v_{i-1}(t) - v_i(t)$. If this value is positive, then the speed of the lead vehicle is greater than that of the following vehicle and the space headway $h_{s_i}(t)$ of vehicle i at time t will increase in magnitude. The opposite is true of a negative relative velocity which results in the space headway decreasing in magnitude. If $a_i(t+\delta t)$ is positive, then vehicle i is accelerating at time $t+\delta t$, whereas on the other hand, if $a_i(t+\delta t)$ is negative, vehicle i is decelerating at time $t+\delta t$. Finally, if $a_i(t+\delta t)$ is zero, then vehicle i is travelling at a constant speed.

Numerous theories and rules based upon the above car-following methodology have been proposed in the literature for governing when and at what rate following vehicles should accelerate or decelerate. Pipes [82] suggested that "a good rule for following another vehicle at a safe distance is to allow yourself at least the length of a car between your vehicle and the vehicle ahead for every 10 miles per hour of speed at which you are travelling." The approach of Forbes [32], on the other hand, considers the reaction time needed for the driver of a following vehicle to perceive the need to decelerate and apply the brakes accordingly, *i.e.* the time gap between the rear of the lead vehicle and the front of the following vehicle should always be equal to or greater than this reaction time. A third example of a car-following theory was the suite of models proposed by General Motors [16, 35, 36, 46]. These were more extensive in comparison to any that had gone before them, largely as a result of their comprehensive accompanying field experiments and the discovery of the mathematical bridge between microscopic and macroscopic traffic flow theories [65]. Five generations of car-following models were developed, all taking the

same form, in which a response was triggered as a result of a stimuli coupled with a degree of sensitivity. This response was represented by the acceleration or deceleration of the following vehicle, while the stimuli was the relative velocity of the lead and following vehicles. The five models differed in their respective representations of the sensitivity parameter.

2.2.3 The dynamics of vehicle delay at signalised intersections

There are numerous methodologies in the literature for determining vehicle delay at signalised intersections from an aggregated, macroscopic point of view. These methodologies are, however, typically inspired by the vehicle-specific delays experienced at a microscopic level. For this reason, the dynamics of the delays are discussed in this section on microscopic traffic flow theory.

Vehicle delay at a signalised intersection may be computed as the difference between the travel time that is actually experienced by a vehicle while passing through the intersection and the travel time this vehicle would have experienced in the absence of the traffic signal control and any resulting vehicle queues [24]. This total delay is typically categorised as deceleration delay, stopped delay, and acceleration delay, as is illustrated in Figure 2.3.

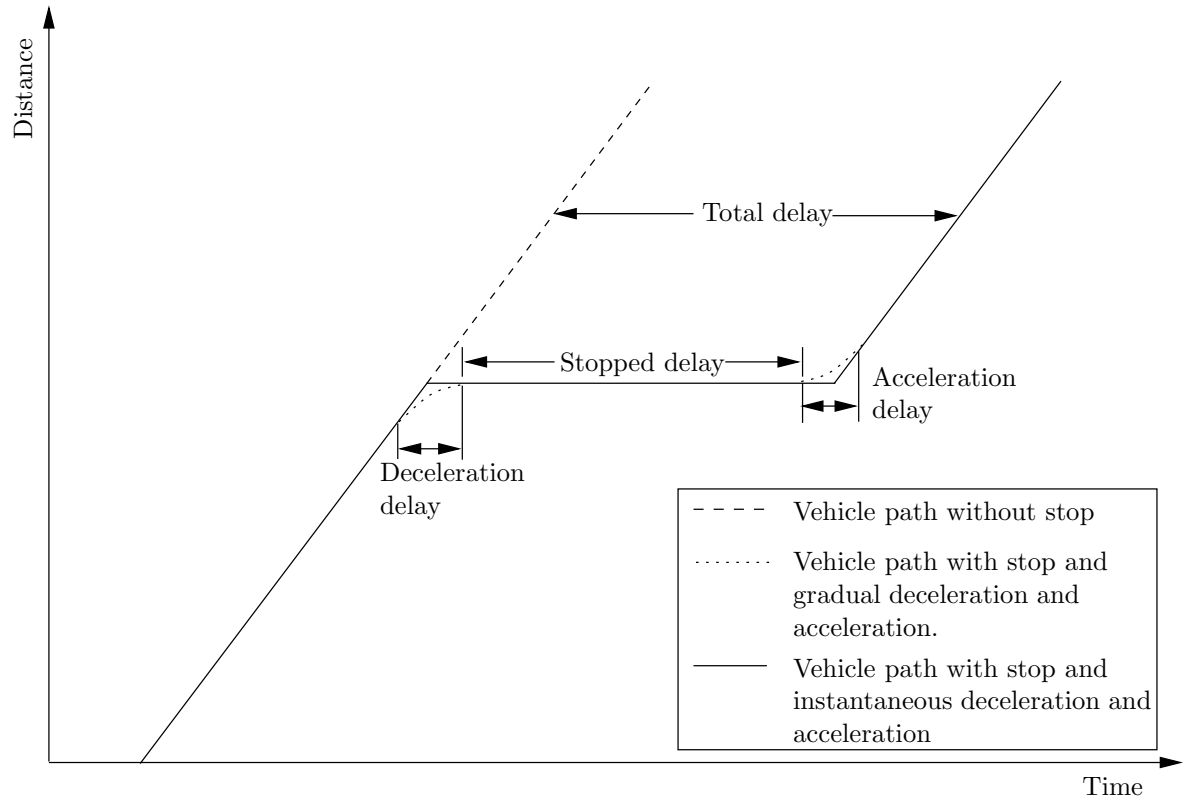


FIGURE 2.3: *Definition of stopped, deceleration and acceleration delay.*

Apart from the delay that is experienced by vehicles waiting for service along an approach to a signalised intersection, an additional delay is also experienced upon the commencement of service as a queue of vehicles departs from the intersection [27]. Ideally, this additional delay would not be prevalent as vehicles begin to discharge from a queue at their desired speeds immediately, following the commencement of a green signal (*i.e.* the vehicles would require no time to react to the signal change and no acceleration is necessary for them to reach their desired speed). This

is a simplifying assumption that is often made in the literature when modelling traffic flow at signalised intersections. In reality, however, this additional delay is attributed to delayed driver reaction times to a signal changing from red to green as well as the time required for vehicles to accelerate to their desired speed.

The dynamics of the delay associated with discharging a vehicle queue upon the commencement of service at a signalised intersection are explained in Figures 2.4 and 2.5. Consider a stationary queue of N vehicles waiting for service at a signalised intersection, as shown in Figure 2.5. It was observed by Greenshields *et al.* [41] that upon departing from the queue, the time headways between the first several vehicles to leave the queue are relatively longer than for the vehicles that follow. These time headways are illustrated graphically in Figure 2.5. The time headway of the first vehicle is the time elapsed between the start of the green signal, and the crossing of the stop line of the intersection of the rear bumper of the vehicle [23] (or any other fixed point on the vehicle, *e.g.* the front wheels are used in [94]). The time headway of each vehicle after the first is the time elapsed between the crossing of the stop line of the intersection of the rear bumper of the vehicle in front of it and its own rear bumper. The time headway of the first vehicle in the queue is comparatively long due to the fact that it has to observe the signal change from red to green, react to this change, and then proceed to accelerate from rest through the intersection. The second vehicle follows the same procedure as the first, but its headway is relatively shorter due to the facts that its reaction time and acceleration time partially overlap with that of the first vehicle and that it has more space in which to accelerate before it reaches the intersection. This pattern continues, with the time headways of each successive vehicle to depart from the intersection becoming smaller, up to some vehicle n (say) after which the effect of driver reactions and finite accelerations has dissipated, as shown in FIgure 2.5. In Figure 2.5, the average time headway $\bar{h}_{t_{\text{sat}}} = \frac{1}{N-n}\sum_{i=n}^{N} h_{t_i}$ achieved by vehicles after the n^{th} vehicle is termed the *saturation time headway*. The saturation time headway is the headway achieved by vehicles that were queued, crossing the intersection at the maximum flow rate possible, Q_{max}. Typically, the time headways of the first n vehicles, on average, exceed $\bar{h}_{t_{\text{sat}}}$, by an amount $t_{h_i} = h_{t_i} - \bar{h}_{t_{\text{sat}}}$. The value of t_{h_i} typically decreases as i approaches n.

It is not a trivial task to calculate precisely the delays caused by driver reaction times and vehicle accelerations and decelerations due to the complexity associated with modelling individual vehicle accelerations and decelerations explicitly. As a result, many techniques have been introduced to account for the aforementioned delays without actually having to model vehicle accelerations and decelerations explicitly. In [1], Allsop provides a review of some of the pioneering expressions for average vehicle delay at a signalised intersection for various arrival and departure processes, as derived by Webster [99], Miller [68] and Newell [74]. In all of these approaches, individual vehicle positions and speeds are not considered, but rather the vehicle arrival rates and saturation flow rates along the roadways approaching the intersection. It is also apparent that the entire cycle length is partitioned into *effective green* and *effective red* times, during which constant traffic characteristics may be assumed (*e.g.* departure rates) rather than considering explicit green, amber and red periods, as is also highlighted in [24]. The effective green time is the portion of a cycle during which vehicles are moving through the intersection, while the effective red time is the amount of time during which traffic along an approach is assumed to be queued or stationary (*i.e.* no vehicles enter the intersection). To account for delay due to driver reactions and finite accelerations it is stated in [1] that the start of the effective green time begins after the signal turns green. The magnitude of this time difference is the sum of the differences between the saturation headway $\bar{h}_{t_{\text{sat}}}$ and the headways of the earlier departing vehicles which experience a headway greater than $\bar{h}_{t_{\text{sat}}}$. The effective red time typically includes the red time and either a fraction or all of the amber time. The above approaches are assumed in [94] when determining the average vehicle delay at signalised intersections analytically.

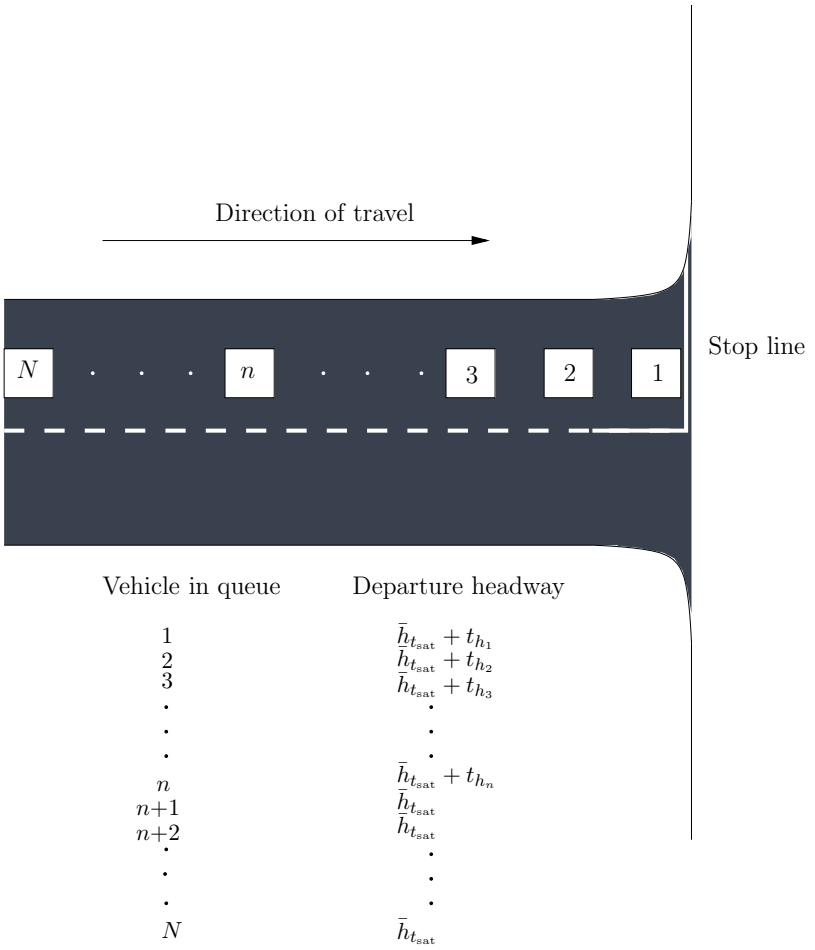


FIGURE 2.4: *A queue of vehicles awaiting service at a signalised intersection with associated headway values.*

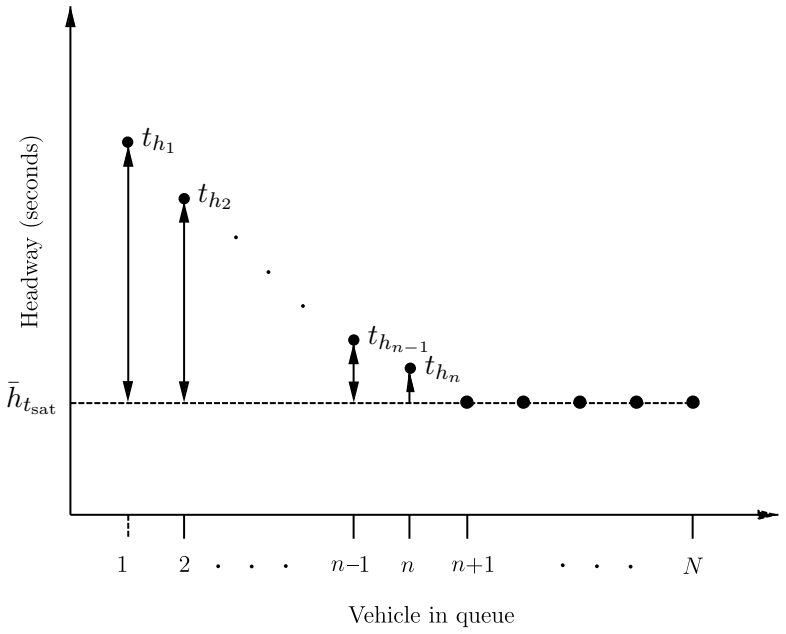


FIGURE 2.5: *Headways of vehicles departing from a queue at a signalised intersection.*

2.3 Macroscopic traffic flow theory

In macroscopic traffic flow theory, multiple vehicles are considered simultaneously as opposed to the more individual vehicle consideration of microscopic traffic flow theory. Instead of considering each individual vehicle in a traffic stream, the traffic stream itself is considered, typically as a fluid [63], whose state is characterised by aggregate macroscopic variables, such as its *density*, its *flow or volume,* and its *speed* [6]. In this section, the traditional definitions of the macroscopic variables of flow (density, occupancy and mean speed) are presented first, and this is followed by their spatio-temporal generalisations. A so-called *continuity equation* and several *fundamental diagrams* are central to macroscopic traffic flow theory. The *continuity equation* describes the relationships between speed, density and flow [78], while the fundamental diagrams are used to describe the statistical relations between the three macroscopic flow variables for various traffic flow conditions [49]. The section closes with brief overviews of these central notions.

2.3.1 Macroscopic traffic flow variables and characteristics

The *rate of vehicle flow* Q_j along road lane j (also known as the intensity or volume of vehicle flow) is considered a temporal measurement [63]. It is traditionally defined as the number N of vehicles that pass a stationary observer during some time interval, divided by the length T of the time interval [21]. According to this definition, it follows that

$$Q_j = \frac{N}{T} = \frac{N}{\sum_{i=1}^{N} h_{t_i}} = \frac{1}{\bar{h}_t}, \tag{2.1}$$

where

$$\bar{h}_t = \frac{1}{N} \sum_{i=1}^{N} h_{t_i}$$

is the average time headway of the N vehicles. The expression (2.1) is a relationship between the flow Q_j and the average time headway $\bar{h}_t$, thereby relating the macroscopic flow variable Q_j to average microscopic time headway variable $\bar{h}_t$.

The *vehicle flow density* K_j of a road lane j is the number of vehicles per distance unit, making it an instantaneous variable (*i.e.* it is computed at a specific time instant) [49]. The density therefore provides a measure of how crowded a certain section of road is. It should, however, be noted that the notion of density does not take into account the effects of traffic composition and varying vehicle lengths, as it only considers the number of vehicles on a road section[2]. The density is calculated as

$$K_j = \frac{N}{L} = \frac{N}{\sum_{i=1}^{N} h_{s_i}} = \frac{1}{\bar{h}_s}, \tag{2.2}$$

where L denotes the length of the road section under consideration and

$$\bar{h}_s = \frac{1}{N} \sum_{i=1}^{N} h_{s_i}$$

[2]To account for heterogeneous vehicle types, the notion of *passenger car units* or PCUs may be introduced, where an average passenger car is the equivalent of one PCU and a larger vehicle may be the equivalent of several PCUs [63].

is the average space headway of the N vehicles travelling on the road section. As with flow, the macroscopic flow variable of density, K_j, is therefore related to the average microscopic variable of average space headway, $\bar{h}_s$, by the expression in (2.2).

Occupancy is a variable of a traffic stream along a road lane which is measured at a particular location and is related to the density of the traffic stream [21]. Occupancy is calculated by multiplying the density of a traffic stream by the average length of the vehicles in the stream. Using (2.2), it follows that

$$O_j = \left(\frac{1}{N}\sum_{i=1}^{N}\ell_i\right)\frac{N}{L} = \frac{1}{L}\sum_{i=1}^{N}\ell_i = \bar{\ell}K_j, \tag{2.3}$$

where

$$\bar{\ell} = \frac{1}{N}\sum_{i=1}^{N}\ell_i$$

denotes the average length of the N vehicles under consideration. From (2.3) it may be seen that occupancy represents the real density of the flow in terms of the physical space occupied by all the vehicles on the road section [63].

The *mean speed* of a traffic stream may be computed in one of two ways, the first being from a fixed cross-sectional point along a road, giving the *time-mean speed* $\bar{v}_t$, or at a time instant, giving the *space-mean speed* $\bar{v}_s$ [63]. These two terms are considered in the following section after having introduced the necessary notation.

2.3.2 Generalised macroscopic traffic flow variables

The introduction of alternative traffic measurement instruments and methods, such as automatic vehicle identification, radar and floating car data, have prompted novel approaches towards determining the values of the macroscopic flow variables described above [49]. This is due to the fact that these instruments are able to provide both spatial *and* temporal information about the different aspects of traffic flows, as opposed to only local or instantaneous information. In [49], the work of Edie [26] is considered central to relating these local and instantaneous variables and generalising the definitions of flow, density, occupancy and mean speed.

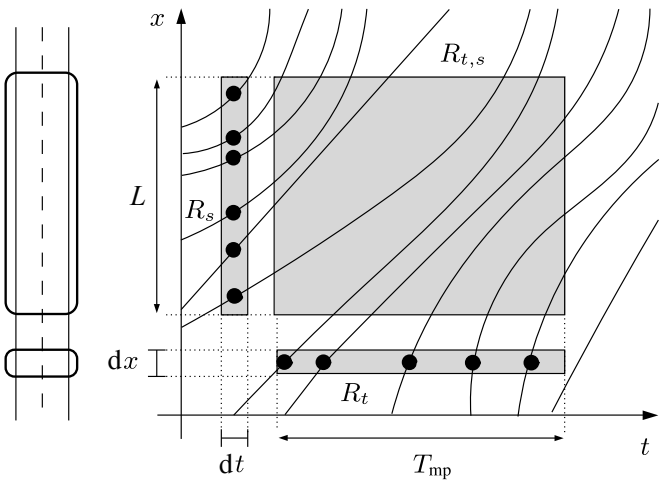


FIGURE 2.6: *Time-space diagram with three measurement regions, R_t, R_s and $R_{t,s}$ (diagram based on [63]).*

Consider the time-space diagram in Figure 2.6. It shows the complete trajectories of several vehicles (including all associated accelerations and decelerations) and highlights three different measurement regions, R_t, R_s and $R_{t,s}$. These rectangular regions are bounded in time and space by a measurement period T_{mp} and a section of road of length L. The black dots represent individual measurements.

The measurement region R_t corresponds to measurements taken at a single fixed point $\mathrm{d}x$ in space, during the time period T_{mp} (*e.g.* provided by a single inductive loop embedded in the road surface). The region labelled R_s corresponds to measurements made at a single instant $\mathrm{d}t$ in time, over a road section of length L (*e.g.* provided by an aerial photograph of a road section). Finally, the measurement region $R_{t,s}$ corresponds to measurements taken in a general spatio-temporal region. This region may take any shape, but is shown for the sake of simplicity as a rectangle in Figure 2.6. An example of this kind of region results from a sequence of image measurements captured by a radar or video camera detector [63].

This general measurement region allows for the redefinition of the four macroscopic flow variables introduced in §2.3.1, starting once again with flow. Flow may be defined as the total distance travelled by all the vehicles along road lane j in the measurement region, divided by the area of this region [21, 26]. Letting X_i be the distance travelled by vehicle i during the time interval $\mathrm{d}t$, this generalisation allows for the flow to be computed using the spatial measurement region R_s as

$$Q_j = \frac{\sum_{i=1}^{N} X_i}{L\mathrm{d}t} = \frac{1}{L\mathrm{d}t}\sum_{i=1}^{N} v_i \mathrm{d}t = \frac{1}{L}\sum_{i=1}^{N} v_i, \tag{2.4}$$

where v_i is again the speed of vehicle $i \in \{1, \ldots, N\}$ and L is the length of the road section under consideration. If consecutive flow measurements are now considered in the region $R_{t,s}$, a formulation corresponding to the temporal average of the flow may be derived. Assuming that at each time step t during the period T_{mp} the flow $Q_j(t)$ along road lane j is known for consecutive values of R_s, the generalised definition of flow is

$$Q_j = \begin{cases} \dfrac{1}{T_{\text{mp}}}\displaystyle\int_{t=0}^{T_{\text{mp}}} Q_j(t)\mathrm{d}t & \text{(continuous)}, \qquad (2.5) \\[2ex] \dfrac{1}{T_{\text{mp}}}\displaystyle\sum_{t=1}^{T_{\text{mp}}} Q_j(t) & \text{(discrete)}. \qquad (2.6) \end{cases}$$

The generalisation of density follows similarly. Considering Figure 2.6 again, density may be defined as the total time spent by all the vehicles in the measurement region, divided by the area of this region [26, 63]. If the travel time and speed of vehicle i are T_i and v_i, respectively, then the density may be computed as

$$K_j = \frac{\sum_{i=1}^{N} T_i}{T_{\text{mp}}\mathrm{d}x} = \frac{1}{T_{\text{mp}}\mathrm{d}x}\sum_{i=1}^{N} \frac{\mathrm{d}x}{v_i} = \frac{1}{T_{\text{mp}}}\sum_{i=1}^{N} \frac{1}{v_i}. \tag{2.7}$$

It may be seen from (2.7) that it is necessary to know the individual travel time T_i of each vehicle $i \in \{1, \ldots, N\}$ travelling along road lane j. This information is not always available and can be difficult to measure. To overcome this obstacle, the temporal average of the density is considered when calculating the density in the region $R_{t,s}$. Once again assuming that at each time step t during the period T_{mp} the density $K_j(t)$ along road lane j is known for consecutive

R_s regions, the generalised definition of density is

$$K_j = \begin{cases} \dfrac{1}{T_{\text{mp}}} \displaystyle\int_{t=0}^{T_{\text{mp}}} K_j(t)\mathrm{d}t & \text{(continuous)}, \quad (2.8) \\ \dfrac{1}{T_{\text{mp}}} \displaystyle\sum_{t=1}^{T_{\text{mp}}} K_j(t) & \text{(discrete)}. \quad (2.9) \end{cases}$$

Most existing traffic detector stations along a road are only capable of temporal measurements in the region R_t [63]. This temporal measurement is the equivalent of the aforementioned occupancy, which corresponds to the time the measurement location was occupied by a vehicle, and is calculated as

$$O_j = \frac{1}{T_{\text{mp}}} \sum_{i=1}^{N} t_{\text{o}_i}, \tag{2.10}$$

where t_{o_i} denotes the so-called *on-time* of vehicle i. This on-time value is the amount of time any part of the vehicle is present above the detector and corresponds to the grey-shaded area swept by the vehicles in Figure 2.1. It is calculated as the effective vehicle length detected divided by the vehicle's speed [17], that is

$$t_{\text{o}_i} = \frac{\ell_i + L_d}{v_i}, \tag{2.11}$$

where ℓ_i is the length of vehicle i and L_d is the length of the detection zone. From here, it is possible to determine the occupancy for the generalised measurement region $R_{t,s}$ by dividing the total space consumed by vehicles in the time-space diagram (such as that shown in Figure 2.1), by the area of the measurement region [15, 21, 26].

The generalised definitions of the two different mean speeds mentioned earlier are now discussed. The space-mean speed may be defined as the total distance travelled by all the vehicles in the measurement region, divided by the total time spent in the region [21, 26]. From this definition, the space-mean speed may be calculated for each of the spatial and temporal regions, R_s and R_t, respectively as

$$\bar{v}_s = \frac{\sum_i^N X_i}{\sum_i^N T_i} = \begin{cases} \dfrac{\sum_i^N v_i \mathrm{d}t}{N\mathrm{d}t} = \dfrac{1}{N}\displaystyle\sum_i^N v_i & \text{(region } R_s\text{)}, \quad (2.12) \\ \dfrac{N\mathrm{d}x}{\sum_i^N \frac{\mathrm{d}x}{v_i}} = \dfrac{1}{\frac{1}{N}\sum_i^N \frac{1}{v_i}} & \text{(region } R_t\text{)}. \quad (2.13) \end{cases}$$

Here, X_i and T_i are respectively the distance travelled and the time spent in the measurement region by vehicle i. This space-mean speed is also known as the *average travel speed*. It may be seen that the spatial measurement is based on the arithmetic mean of the vehicles' instantaneous speeds while the temporal measurement is calculated using the harmonic average of the vehicles' spot speeds. A similar approach is taken to calculate the *time-mean speed* $\bar{v}_t$. The only difference is that the arithmetic mean of the vehicles' spot speeds is used to calculate $\bar{v}_t$ for the temporal region R_t, while the harmonic average of the vehicles' instantaneous speeds results in the time-mean speed for the spatial region R_s.

Wardrop [96] showed that the relation

$$\bar{v}_t = \bar{v}_s + \frac{\sigma_s^2}{\bar{v}_s} \tag{2.14}$$

holds between the time-mean and space-mean speeds, where σ_s^2 is the statistical sample variance, calculated as

$$\sigma_s^2 = \frac{1}{N-1}\sum_{i=1}^{N}(v_i - \bar{v}_s)^2, \tag{2.15}$$

and v_i is the instantaneous speed of vehicle i. The difference between $\bar{v}_s$ and $\bar{v}_t$ is often negligible in lighter, free-flowing traffic conditions, but under more congested conditions this difference can become substantial [63].

2.3.3 The continuity equation of macroscopic traffic flow theory

There exists an important relationship between density, flow and space-mean speed, called *the fundamental relation of traffic flow theory* [96]. This relationship is also known as *the continuity equation* [49] and is given as

$$Q_j = K_j \bar{v}_s. \tag{2.16}$$

It has been shown that no similar relationship holds for the time-mean speed [65]. The importance of (2.16) is that, given any two of the three macroscopic variables, the equation allows for the third to be calculated [63]. This consequence is particularly important in the case of density, which is not always easily measured [63].

2.3.4 The fundamental diagrams of macroscopic traffic flow theory

Three fundamental diagrams are presented in this section which provide graphical representations of the statistical relationships between the macroscopic traffic flow variables of flow, density and speed for different traffic flow conditions and are based on the premise that under similar traffic conditions, drivers will behave similarly [49]. In [63], three categories of traffic flow conditions are distinguished, namely *free flow traffic, capacity-flow traffic* and *congested traffic.*

Free-flow traffic typically occurs under light traffic flow conditions when vehicles are able to travel at their desired speeds, unimpeded by slower moving vehicles or vehicle queues [63]. These *desired* or *free-flow* speeds are influenced by characteristics of the vehicles, the drivers and the road section in question, as well as by weather conditions and traffic rules (*e.g.* speed limits) [49]. These desired speeds are summarised by the mean speed of all the vehicles on a roadway, each travelling at its own desired speed, and is denoted by $\bar{v}_0$. Under free-flow traffic conditions large average space headways are typically observed due to low densities and, as a result, small local disturbances in the temporal and spatial patterns of the traffic stream (*e.g.* overtaking manoeuvres or sudden braking) have no significant effects, and hence the traffic flow is considered *stable* [63].

As traffic density increases, so too does traffic flow, while the spacing between vehicles decreases. At a certain point, the traffic flow along road lane j will reach a maximum value, which is determined by the mean speed of the traffic stream as well as its current density, known as the *capacity-flow* and denoted by $Q_j^{\max}$. From (2.1) it may be seen that the average time headway is minimal at capacity-flow traffic, indicating the formation of tightly packed clusters, or platoons of vehicles, travelling at the *capacity-flow speed* $\bar{v}_c$ which is typically slightly lower than the free flow-speed [63]. These platoons are often unstable, however, with even the slightest slowing down of a vehicle having a backward cascading effect resulting in exaggerated braking by following vehicles.

As vehicle density increases further along road lane j, vehicles eventually have to start braking to avoid collisions due to too small time and space headways, with the resulting chain reactions of following vehicles leading to a breakdown in flow. The resulting saturated traffic conditions are known as congested traffic [63]. The density at which these breakdowns occur is described as moderately high [63] and is called the *critical density* K_j^{crit}. Further increases in vehicle density above the critical density can result in what is known as *stop-and-go traffic* in which vehicles are required to slow down severely or even stop. When traffic does become motionless, the space headway between all vehicles reduces to a minimum bumper-to-bumper distance, in which case the traffic state is described as *jammed traffic*. The maximum density at which traffic becomes stationary along road lane j is called the *jam density*, denoted by K_j^{jam}.

The relationship between space-mean speed and density of road lane j is illustrated in Figure 2.7. From the figure it may be seen that the density K_j has extremal values of 0 and K_j^{jam}, while the space-mean speed varies between zero and the desired free-flow speed $\bar{v}_0$. It may also be observed that as the density increases, the space-mean speed decreases.

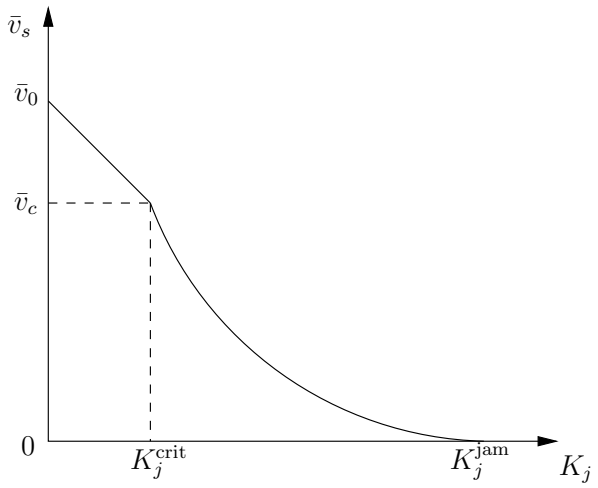


FIGURE 2.7: *A fundamental diagram relating density to the space-mean speed.*

The relationship between traffic flow and density along road lane j is illustrated in Figure 2.8. From the figure it may be seen that the capacity flow $Q_j^{\max}$ is reached at the critical density K_j^{crit}. Also noticeable in Figure 2.8 is the fact that the space-mean speed may be defined for any point on the curve as the slope of the line through that point and the origin. In the free-flow region of Figure 2.8 (*i.e.* for densities lower than K_j^{crit}) the flow increase is almost linear, while in the congested region (*i.e.* for densities greater than K_j^{crit}) the flow degrades continuously until the jam density K_j^{jam} is reached, at which point flow becomes zero.

The relationship between the space-mean speed and the vehicle flow along road lane j is illustrated in Figure 2.9. This figure is not as easily interpreted as the previous two due to the fact that two space-mean speed values are associated with each flow value. The free-flow region of the diagram lies above the horizontal line through $\bar{v}_c$ which represents the capacity-flow speed, or the speed at which the flow is maximal, *i.e.* $Q_j = Q_j^{\max}$. Points along the curve in this free-flow region indicate fewer vehicles passing a fixed point at higher speeds with larger space headways separating them, while the corresponding point along the curve in the congested region of the diagram (*i.e.* below the line through $\bar{v}_c$) is indicative of a larger number of vehicles passing the same fixed point at slower speeds with smaller space headways separating them.

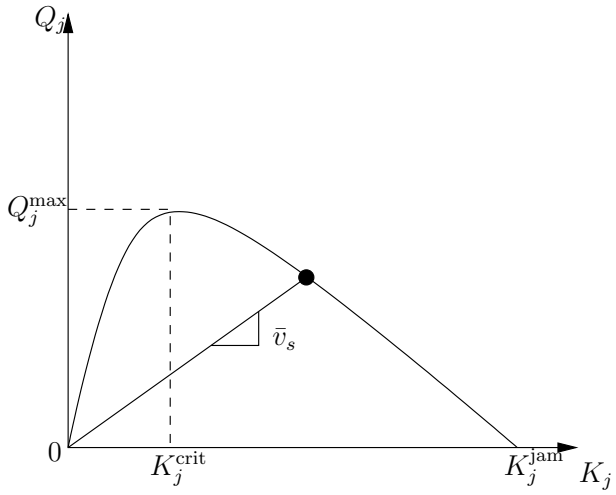


FIGURE 2.8: *A fundamental diagram relating density to flow.*

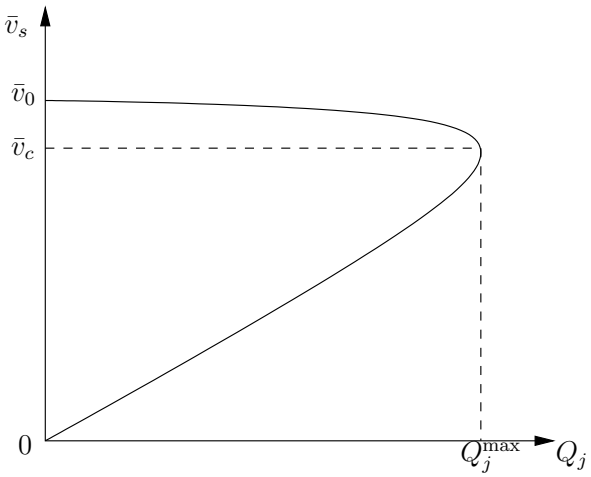


FIGURE 2.9: *A fundamental diagram relating flow to the space-mean speed.*

2.4 Chapter summary

A brief review of traffic flow theory was presented in this chapter. A distinction was made between the theory associated with microscopic and macroscopic traffic flows. In microscopic traffic flow theory, the focus is on each individual vehicle in a traffic stream and its associated characteristics, while in macroscopic traffic flow theory, the focus is on the aggregated flow of traffic. The concept of car-following models was introduced in the section on microscopic traffic flow theory. After having introduced the basic macroscopic traffic flow variables, expressions were presented illustrating the relationships between macroscopic and microscopic traffic flow variables. The chapter closed with the presentation of the fundamental relation of traffic flow theory as well as a description of three prevalent categories of traffic flow volumes and the effects that they have on the behaviour of and relationships between the macroscopic traffic flow variables.

CHAPTER 3

Computer simulation modelling

Contents

This chapter contains a description of simulation modelling from the literature as well as the principles of simulation modelling in §3.1. The concepts and components of a simulation model, the various categories of simulation modelling, and various prevailing simulation modelling paradigms are discussed in §3.1.1, §3.1.2 and §3.1.3, respectively. The steps followed in a typical simulation study are described in §3.2. This is followed by a discussion on the advantages and disadvantages of employing simulation as an analytical tool in §3.3. In §3.4, attention is focused on traffic simulation modelling. Mico-, meso-, and macroscopic traffic simulation models are discussed in §3.4.1, §3.4.2 and §3.4.3, respectively, as well as examples of commercially available traffic simulation software packages which harness them.

3.1 Principles of simulation modelling

There are numerous interpretations in the literature of what simulation is and how it should be defined. It has been proposed that simulation is a technique which permits the study of a complex system in a laboratory rather than in the field [25]. It has also been suggested that simulation is a dynamic representation achieved by building a model and moving it through time [66]. A more involved definition is that the simulation of a system is the operation of a model that is a representation of the system, is amenable to manipulations, and from which properties concerning the behaviour of the actual system can be inferred [73]. A simulation model is described by Banks *et al.* [5] as the imitation of a real-world process or system over

time such that the behaviour of the system can be studied. If the model is a sufficiently realistic imitation of the real-world process, then data may be collected from this model as if it were collected directly from the real system under observation. Banks [5] goes on to state that simulation is used to describe and analyse the behaviour of a system, allowing for "what-if" questions to be asked about the real-world system, as well as aiding in the design of real-world systems (as simulation allows for both existing and conceptual systems to be modelled). Over time, simulation models have become extremely useful — almost indispensable, in fact — when analysing and verifying theoretical models which may be too difficult to analyse on a purely conceptual level [19].

When developing a simulation model, several questions are raised concerning the model to be built and analysed [5]. Examples of these questions include:

1. How is the form of the input data determined?
2. What statistical distribution must be used to generate random variates?
3. Does the model adequately imitate reality?
4. How long does the simulation model require to run?
5. How many simulation iterations must be conducted?
6. What statistical techniques should be used to analyse the model output?

To be able to provide answers to these questions, it is important to acquire an understanding of the problem at hand and the goal that is to be achieved by the simulation model. Before the model building process begins, it is required that potential areas of concern are identified and analysed. One such example is the availability of information pertaining to the system to be modelled, as well as the ease with which data may be obtained. In other words, are the data readily available, or would data collection be necessary and if so, how complicated are the collection procedures [9]. Also considered to be of importance is the identification of fundamental system characteristics and the ability to distinguish them from insignificant, negligible system attributes. Most importantly, the validity and integrity of the model must be considered in terms of its ability to provide an accurate account of the real-world system being modelled while also producing viable and accurate results [27].

The above factors influence the level of detail required in a simulation model. Law [55] considers the selection of an adequately and appropriately detailed simulation model to be an art in itself. This selection of the required level of detail has a direct influence on the selection of the simulation modelling paradigm to be employed to represent the real-world system under consideration [9]. Three different levels of abstraction are described in [9] and [102], namely the *strategic level*, the *operational level*, and the *physical level.*

Simulation studies performed on a strategic level (also referred to as a *macro level*) assume a high level of abstraction and are typically employed to identify and analyse strategic organisational aspects of the real-world system, requiring relatively little detail. Examples include the simulation of the effectiveness of an advertising campaign, or the spread of a disease among a population of potential carriers [27].

A simulation model built at the operational level (also referred to as the *meso level*) is relatively less abstract than one built at a strategic level. As a result, more detail is incorporated into this type of model as it is typically employed to make tactical decisions about the real-world system it is replicating [27]. Examples of problems for which an operational level simulation model

may be applicable include the investigation of the optimal inventory levels in a warehouse, the balancing of production lines in a factory and the identification of problems in a system through analysis of relative performance measures.

The most detailed simulation models are built on a physical level (or *micro level*). This very detailed level of abstraction is required when it is necessary to observe and analyse the behaviour and characteristics of each individual entity of the system under investigation, in terms of their dimensions, speeds, positions and timings [9]. Such detail is typically required when studying the dynamics of a transportation network or the movement of passengers through a bus or subway terminal, or when investigating evacuation procedures and their effectiveness.

3.1.1 The concepts and components of a simulation model

Several underlying concepts and components of a *system* have to be recreated in a simulation model. These include the *entities* of the system and their associated *attributes* and *activities*, as well as the *events* that occur which may result in changes to the *system state variables*, and thus the *system state*.

A *model* may be defined as a representation of a *system* for the purpose of studying the system, and a system may be defined as a collection of *entities* that interact with one another over time to accomplish one or more goals [2]. The model incorporates only those components which are deemed relative to the study at hand and the level of abstraction required.

An *entity* is an object of interest in the system (*e.g.* a person or vehicle). An entity may be dynamic, possessing the ability to move through a system, or may be static, in which case it will typically interact with or serve other entities in the system. Associated with each entity are one or more *attributes*. An *attribute* represents a characteristic or property unique to that entity [2, 5, 27].

The *state* of a system is the collection of variables necessary to describe the system at any time, relative to the objectives of the study [2]. These variables are referred to as *system state variables* and contain all information required to define what is happening within the system at a given time instant [5].

An *activity* is defined as a period of time with a specified duration, which is known prior to its commencement [2, 5]. This duration may be constant or a random variable drawn from a statistical distribution. A *delay*, on the other hand, is defined as a period of time of indefinite, unknown duration which is incurred as a result of some combination of system conditions [2, 5]. The beginning and the end of an activity or delay are known as *events*.

An *event* is therefore considered an instantaneous occurrence which changes the state of the system [2], and may be *endogenous* or *exogenous* in nature. An endogenous event occurs within the system while an exogenous event is the result of some external influence on the system [27].

3.1.2 Types of simulation models

Banks *et al.* [2] classify simulation models as being static or dynamic, deterministic or stochastic, and discrete or continuous.

A *static* simulation model (also referred to as an analytical model in [9]) represents a real-word system at a particular point in time [2]. The results generated by such a model are functionally dependent on the input parameters. *Dynamic* simulation models, on the other hand, represent real-world systems as they change and evolve over time.

A *deterministic* simulation model contains no random variables, *i.e.* all parameter inputs to the system are known values which result in a unique set of outputs [2]. A *stochastic* simulation model, on the other hand, contains at least one random variable input value. As a result, the output generated by the model is itself random, and may therefore be considered as a statistical estimate of the true characteristics of the system being modelled.

A *discrete* simulation model is one in which the all model variables are updated at discrete, instantaneous points in time. In contrast, a *continuous* simulation model is one in which model variables change continuously over time [2]. It is noted, however, that few simulation models are either wholly discrete or wholly continuous [55], but are rather a combination of the two. Indeed, there exist *discrete-continuous* simulation models which comprise both discrete and continuous variables [5]. In such a model, the values of the system state variables are computed continuously, while the values of attributes of entities and global variables are calculated at discrete points in time [27].

3.1.3 Simulation modelling paradigms

When building a simulation model, there are four main distinguishable paradigms (or approaches) of simulation modelling which may be followed, depending on the level of abstraction and detail required in building the model. These are *system dynamics modelling, discrete event modelling, agent-based modelling* and *dynamic systems modelling* [9, 102].

System dynamics modelling

John Sterman [90] describes system dynamics modelling as "a perspective and set of conceptual tools that enable us to understand the structure and dynamics of complex systems." System dynamics modelling typically assumes a relatively high level of abstraction and is typically employed to study the information-feedback characteristics of activities to show how organisational structure, amplification in policies, and time delays in actions and decisions interact to influence the performance of the system [9]. In a system dynamics model, real-world processes are represented in the form of stocks (*e.g.* material, knowledge or money), flows between these stocks, and information that determines the values of these flows [102]. System dynamics modelling is typically used in long-term, strategic models. Due to the high level of aggregation assumed, people, products, events and other discrete entities are represented by their associated quantities and therefore lose any individual properties, histories or dynamics [9, 102].

Discrete event modelling

In discrete event modelling, system state changes occur at a countable number of discrete points in time, with the goal of portraying the activities in which the entities of the system engage so as to gain insight into the system's dynamic behaviour. This requires clear definitions of the system's various states as well as the activities and events that are responsible for the transition of the system from one state to another. The dynamics of the system are made evident as the simulation model time advances from one event to the next, but the state of the model remains constant between consecutive events. [27].

Agent-based modelling

Agent-based modelling is defined as an essentially decentralised, individual-centric approach to model design [102]. The fundamental difference between agent-based modelling and system dynamics or discrete event modelling is that in the case of agent-based modelling, the behaviour of the global system as a whole is not defined, but rather the behaviour of the individual entities or agents of the system (*e.g.* people, vehicles or companies), and it is from the interactions among and between these agents that the global behaviour of the system emerges [9, 27].

Dynamic systems modelling

Dynamic systems modelling is considered to be the "ancestor" of system dynamic modelling and is an integral part of design processes in mechatronic, electrical, chemical, and other technical engineering disciplines [102]. The underlying mathematical model of a dynamic systems model consists of a number of state variables and various algebraic and/or differential equations pertaining to these variables.

The four modelling paradigms described above are applicable to models of differing levels of detail, complexity and abstraction, as required. It is, however, possible that an integration of the various paradigms occurs in the form of a multi-paradigm model architecture [9].

3.2 Steps in a typical simulation study

The implementation of a successful simulation study typically requires twelve steps to be followed (some of which may need to be repeated). These steps are outlined in [2, 5] as well as in numerous other simulation texts. The twelve steps are discussed individually in this section and are summarised graphically in Figure 3.1.

1. *Problem formulation.* The first step in a simulation study is defining the problem at hand by means of a formal problem statement [2, 5]. This statement must clearly define the overall objectives of the study and must include all relevant questions that are to be answered by the end of the study so as to enable one to decide on the required level of model detail and simulation modelling approach to be followed [55].

2. *Project planning.* The project plan serves to highlight the scope of the project and outlines the various subsystems to be investigated and the scenarios in which they will be investigated. The project plan should also include information pertaining to the performance measures and methods to be used in order to evaluate and compare the different system configurations investigated. A general plan for the study should be provided in terms of expected completion times, costs, requirements and outputs of each stage of the study.

3. *Model conceptualisation.* During this step the real-world system under investigation is abstracted by a conceptual model consisting of a series of mathematical and logical relationships pertaining to the components and structure of the system [5]. The art of modelling lies in the ability to abstract only the essential features of a problem, to select and modify basic assumptions that characterise the system, and then to enrich and elaborate the model until a useful approximation results [2]. For this reason, it is recommended that the modelling of the real-world system begins simply and that the model grows until a model of appropriate complexity has been developed. When deciding on this appropriate

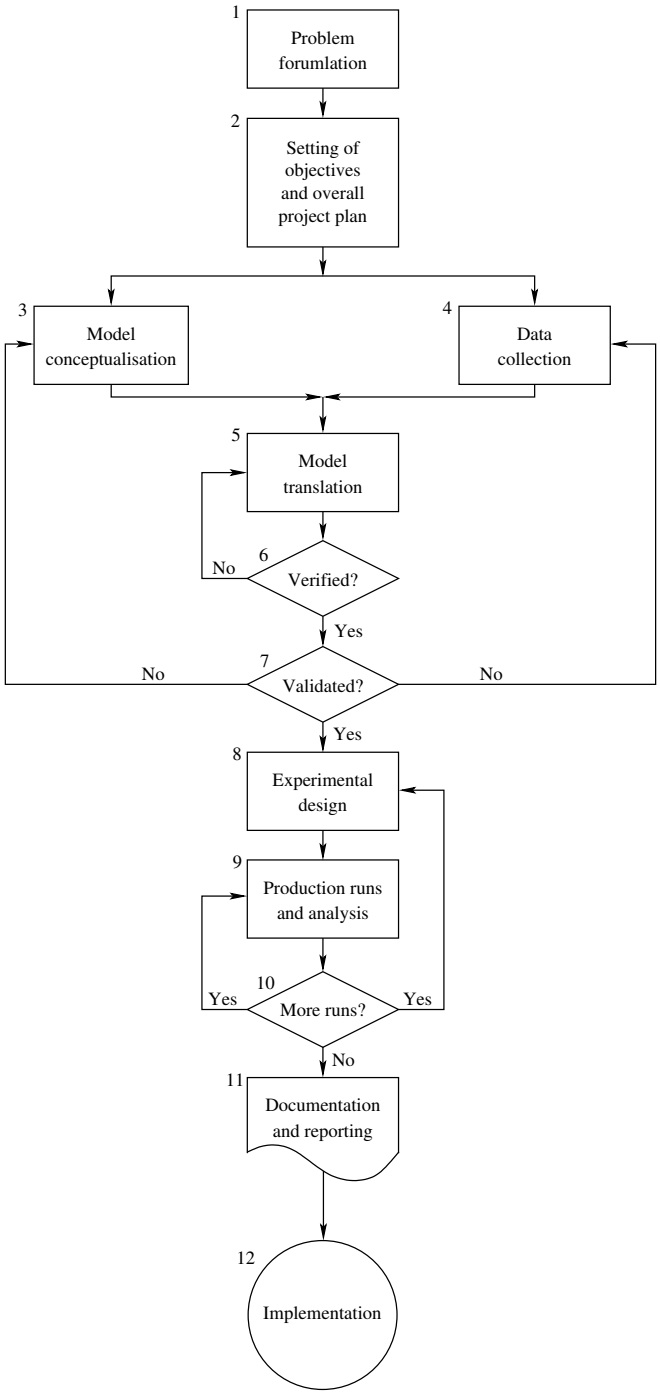


FIGURE 3.1: *The twelve steps in a typical simulation study.*

level of model complexity it is useful to consider again the project scope and objectives, the availability of data, and the intended performance measure indicators. Furthermore, the constraints of the computer system on which the model will be implemented also has to be considered [55].

4. *Data collection.* As may be seen in Figure 3.1, data collection occurs concurrently with model conceptualisation. In fact, there is a constant interplay between the construction of the model as a whole and the collection of required input data, with the objectives of

a study dictating, to a large extent, the required data to be collected [2]. Also, as the complexity of the model changes, so the data requirements may change too. It is also important to note that while collecting the required input data, it is also necessary to collect data pertaining to the performance of the current real-world system for validation purposes [55].

5. *Model translation.* This step requires the selection of a suitable programming language or simulation software package which will be used to convert the conceptual model developed in Step 3 above into an operational model [2]. The benefit of using a known programming language to code and implement the conceptual model is that it gives the user greater control over and understanding of the model, while using a specialised simulation software package may greatly reduce programming time and thus also the time to project completion [55].

6. *Model verification.* Once the model has been translated, it is necessary to verify whether or not the simulation model is performing as expected. It is advised that verification should occur on a continual basis throughout the construction of the model [2, 5]. In some instances, interactive run controllers and debuggers may be employed to aid in the verification process.

7. *Model validation.* Validation is the process of determining whether the conceptual model is an adequate representation of the real-world system. Validation is typically achieved through an iterative process. The output of the simulation model is compared to that of the real-world system, and, if there are any discrepancies, these are used together with any inputs from subject matter experts to recalibrate the model in an attempt to improve its accuracy. This process is repeated until the accuracy of the model is found to be acceptable [2].

8. *Experimental design.* During this step the alternative system designs which are to be investigated are decided upon. For each scenario in which the systems will be investigated, a decision is made pertaining to the duration of the simulation run, the number of runs or replications to be implemented, and the length of the model initialisation period [2, 5].

9. *Production runs and analysis.* Production runs and their subsequent analysis are used to estimate measures of performance for the system designs that are being simulated.

10. *More production runs.* Based on the output of the initial production runs, it is decided whether or not additional production runs are required. If so, the design of these additional experiments should follow.

11. *Documentation and reporting.* Documentation refers to both documentation of the program used to implement the simulation model as well as documentation of the progress of the simulation study. Program documentation is necessary if the program is to be used by other analysts outside of the particular study. In addition, if the program is to be modified in any way in future, then accurate documentation may greatly facilitate this process. Progress reports serve to provide a chronology of the work done and the decisions made throughout the project. It is recommended that frequent reporting take place throughout the study [71]. This progress documentation should include information on model specifications, prototype demonstrations, model animations, and progress reports [5].

12. *Implementation.* The final implementation of the processes suggested by the simulation study depends on how successfully the previous eleven steps have been implemented and how convincing the findings of the study are to the final decision maker [2].

3.3 The advantages and disadvantages of simulation modelling

Advancements in the information technology sector have led to the development of improved, more powerful computational hardware. This, in turn, has had a positive effect on the simulation software industry as simulation software platforms are now able to harness this improved hardware performance, enabling more accurate executions and quicker completion times, allowing for the expansion of the application of simulation to a wider variety of more complex systems in industry [5].

One of the primary advantages of simulation modelling is that it allows for the control of a system in its entirety while providing users with practical feedback during the design of real-world systems [19]. This allows for the implementation and investigation of novel system design strategies and policies in an attempt to gauge their effectiveness and correctness before any real-world implementation takes place and without committing any additional resources [5]. Simulation modelling also allows for modifications or additions to currently employed processes to be made without disrupting ongoing real world systems [2]. As a consequence, numerous proposed system designs may be compared using simulation in an attempt to ascertain which one best meets the specified system requirements. All of the above factors make simulation modelling a comparatively economical alternative to direct real-world implementation. In fact, the typical cost of a simulation study is typically substantially less than 1% of the total amount expended for the implementation of a design or redesign of a real-world system [5].

A second noteworthy advantage of simulation models is that they afford the user the ability to expand and compress time. The simulation model time may be slowed so as to allow the user to investigate phenomena which occur too rapidly to analyse efficiently in real time, and analogously, the simulation model time may be increased such that the system's evolution over a number of days or even weeks may be observed in a matter of minutes.

Craig [19] considers the ability to study systems from several levels of abstraction as an advantage of simulation modelling. By studying a system from a higher level of abstraction, one is better able to understand the behaviours and interactions of the high-level components of the system, thereby counteracting the complexity of the actual real-world system.

Many simulation software packages allow for an animation to accompany the simulation model while it runs, facilitating visualisation of the system and how its various components operate and interact [5]. This aids significantly in the verification and validation of the model and is useful for informing and inspiring alternative system designs as it is able to reveal information about the model which mere results may not be able to achieve.

A further advantage of simulation modelling is the scale of problems which may be implemented in a simulated environment. Simulation allows for large scale, complex problems with significant levels of realism to be implemented and executed relatively efficiently. Similar problems would be very challenging to implement in an analytic framework, say, while attempting to generate accurate and credible results.

Finally, simulation modelling allows for bottleneck analysis to be performed, indicating where work-in-process, information, materials, *etc.* are being delayed excessively [80].

In spite of these advantages simulation modelling is, however, not without its disadvantages. One such disadvantage is that the building of an accurate and credible simulation model is an art as much as it is a science and requires a great deal of skill and often specialised training [2, 5]. As building a simulation model requires skill, so too does the interpretation of the results and data generated by the model. The impacts of these two disadvantages have been lessened to a

certain extent over time by the introduction of user-friendly simulation software packages, many of which include ready-to-implement models as well as data analysis tools and functionality. A drawback of these simulation packages, however, is that they can be expensive. This expense is often outweighed by the analytic benefits of the simulation model itself, but software vendors have started releasing various versions of software packages, ranging from basic to professional, which reduce the required capital outlay if a professional package is not entirely necessary [2, 5]. Simulation modelling, in some instances, may be very time consuming [2, 5]. This may be attributed to the computationally intensive processing required by some simulation models, and, as a consequence, the results of the simulation may not be readily available immediately after the simulation model has started. These delays may be due to an exceedingly large number of entities being simulated or due to complex interactions occurring between entities [19]. This disadvantage is offset somewhat by the previously mentioned advances in computer hardware which facilitate rapidly running scenarios [5], as well as software advancements in the form of pre-existing simulation model libraries being built into many simulation software packages [2].

3.4 Traffic simulation modelling

May [65] defines simulation in the context of traffic flow and control as "a numerical technique for conducting experiments on a digital computer, which may include stochastic characteristics, and involve mathematical models that describe the behaviour of a transportation system over extended periods of real time." Indeed, traffic simulation models allow for a traffic engineer or analyst to generate various traffic scenarios, optimise traffic control strategies, and predict traffic network behaviour at an operational level. Furthermore, they are able to provide an overall picture of the traffic in a network, giving the traffic engineer or analyst the ability to assess current problems and project possible solutions immediately [10]. For example, if a municipality was considering whether or not to add an additional lane to a segment of highway, simulation modelling would allow for the associated benefits of the additional lane to be measured before a decision is taken as to whether or not it should be built. Similarly, if new traffic signal control strategies were to be introduced, they could be tested extensively in a simulated environment before any capital is invested in their installation and implementation.

Numerous traffic simulation models have been developed and introduced since the first simulation of a traffic intersection was undertaken by the Road Research Laboratory in the United Kingdom in 1951 [65]. Today, there are several alternatives for classifying the various traffic simulation models available. Gibson [39], classifies traffic simulation models according to their area of application (*i.e.* intersections, arterials, urban networks, freeways and freeway corridors). Other methods classify traffic simulation models according to the *uncertainty content* of the model, determined by whether it is deterministic or stochastic in nature, and the *time horizon* of the model, which represents the static or dynamic properties of the model [78]. The most commonly accepted method of classification, however, classifies a traffic simulation model according to its level of abstraction. Following this method, there are three distinct classes of traffic simulation models, namely *macroscopic, mesoscopic* and *microscopic models*. There are numerous examples of traffic flow simulation models in each of these three classes. In [10], sixty five microscopic traffic simulation models, sixteen macroscopic traffic simulation models and three mesoscopic traffic simulation models were reviewed and evaluated.

In this section, each of the three aforementioned classes of traffic simulation models are considered separately, together with examples of commercially available traffic simulation models which reside in these respective classes.

3.4.1 Macroscopic traffic simulation

In macroscopic traffic simulation modelling, many of the fundamental theories, variables and characteristics described in §2.3 are translated into simulation models. As a result of this translation, macroscopic traffic simulation models are typically modelled from an aggregated point of view, employing continuum models (such as the continuity equation described in §2.3.3) [57], which are based on a hydrodynamic analogy and regard traffic flows as a particular fluid process whose state is characterised by aggregate macroscopic variables such as density, volume and speed [6]. In their simplest form, these continuum models do not explicitly consider the effects of vehicle acceleration and inertia, and are unable to describe the dynamics of non-equilibrium traffic flow with adequate precision [67]. More advanced, complex variations of this continuum model take into account the effects of vehicle acceleration and inertia by incorporating a *momentum equation*. These momentum equations attempt to account for the dynamic speed-density relationships observed in real-world traffic flow [78].

Macroscopic traffic simulation software

TRANSYT/10 is an off-line program for determining and studying optimal fixed-time, coordinated traffic signal timings in any network of roads for which the average traffic flows are known [18] and is widely accepted around the world as the standard off-line method for setting fixed-time traffic signals [10]. The original *Traffic Network Study Tool*, or TRANSYT, was first developed by Dennis Robertson at the Transportation Road Research Laboratories (UK) in 1967 [78]. TRANSYT/10 is designed to model traffic behaviour and produce fixed-time signal plans that minimise vehicle delay and the number of stops in an urban network of coordinated traffic signals through the optimisation of signal offsets and green time allocations [10]. Individual vehicles and their associated characteristics are not represented explicitly in TRANSYT/10; instead, all calculations are based on average vehicle flow rates, turning movements and queue lengths [78].

3.4.2 Mesoscopic traffic simulation

Mesoscopic traffic simulation models exhibit characteristics of both macroscopic and microscopic traffic simulation models. For example, macroscopic models do not typically simulate lane-changing, merging and diverging behaviour, although a traffic simulation package called KRONOS, which is largely considered a macroscopic traffic simulation model, does simulate these behaviours [78]. Analogously, INTEGRATION is a microscopic traffic simulation model in the sense that individual vehicles are simulated, but the aforementioned lane changing, merging and diverging behaviours are not implemented [78]. Generally, mesoscopic traffic simulation models are primarily used for traffic assignment purposes, although there are exceptions.

Mesoscopic traffic simulation software

The Simulation and Assignment of Traffic in Urban Road Networks (SATURN) [43] model is an example of a mesoscopic traffic simulation model. SATURN employs two distinct phases: the first is a detailed simulation phase of intersection delays and the second is an assignment phase which determines the routes to be followed by origin-destination pairings [43]. The simulation model iterates between the two phases, with the first simulation phase determining flow-delay

curves which are passed on to the assignment phase in which the curves are used to determine route choices and turning movements [78].

3.4.3 Microscopic traffic simulation

Typical microscopic traffic simulation models rely on microscopic traffic flow theory and the car-following models described in §2.2. Microscopic traffic simulation models incorporate many of the characteristics and variables associated with individual vehicles comprising traffic streams, such as speed, size, acceleration, and lane changing behaviour. Microscopic traffic simulation models are typically stochastic in nature and employ a Monte Carlo process to generate random numbers for representing the driver/vehicle behaviour of real-world traffic conditions [78].

Microscopic traffic simulation software

One such example of a microscopic traffic simulation model is VISSIM [31]. VISSIM (a German acronym for *Traffic In Towns: Simulation*) is a microscopic, behaviour-based multi-purpose traffic simulation model which may be used to analyse and optimise traffic flows [27]. It is used worldwide within consultancies and industry, public agencies, and academic institutions [30]. The system architecture of VISSIM comprises two separate models [31]. The first is the traffic flow model and the second is the traffic signal control model. Vehicle detector values are sent from the traffic flow model to the signal control model which uses these detector values to adjust the traffic signals accordingly. The simulation is microscopic, continuous and stochastic in nature [31], rendering an online animation of the traffic flow, and provides offline reports of travel time and delay time distributions. Some common areas of application of VISSIM include, but are not limited to, the development and analysis of management strategies on motorways, corridor studies on arterials with signalised and non-signalised intersections, analyses of alternative actuated and adaptive control strategies in traffic networks, and investigations with respect to so-called traffic calming schemes [30].

3.5 Chapter summary

In this chapter, the general principles of simulation modelling were discussed in §3.1. This discussion included the components and concepts of a simulation model, the different types of simulation models available and the prevailing simulation modelling paradigms. In §3.2, twelve recommended steps for carrying out a successful simulation study were outlined, while the advantages and disadvantages associated with using simulation modelling were discussed in §3.3. Simulation in the context of traffic flow and control was briefly reviewed in §3.4. This review included macroscopic, mesoscopic, and microscopic traffic simulation models as well as examples of commercially available software packages of each.

CHAPTER 4

A microscopic traffic simulation modelling framework

Contents

A microscopic traffic simulation modelling framework is described is described in this chapter. This framework was built for the purpose of the study considered in this dissertation. The framework was built in the Java-based simulation suite *Anylogic 6.8* [101], and allows for the implementation of microscopic traffic simulation models of varying size and complexity. Brief descriptions of the framework, as well as the methods and procedures followed in the construction and implementation of the various components of the framework, are provided in §4.1. The output provided by the models implemented within the framework is discussed in §4.2, while verification and validation of the framework are carried out in §4.3. The chapter closes with a brief summary in §4.4.

4.1 Traffic simulation modelling framework

A microscopic traffic simulation modelling framework was designed and implemented for the purpose of comparing the efficacies of the various novel and pre-existing traffic signal control algorithms presented and reviewed in this dissertation. This framework facilitates the construction of microscopic traffic simulation models in a customisable manner [29]. According to the terminology introduced in §3.1, these microscopic traffic simulation models are built on a physical, or micro level of abstraction. Furthermore, they are dynamic, stochastic and continuous in nature. The simulation modelling approach they employ is that of an agent-based paradigm.

The entities of the model include the individual vehicles which populate the road network, the traffic signals which control the flow of vehicles at road intersections and the road sections on which the vehicles travel. The vehicles are examples of dynamic entities while the traffic signals and road sections are static entities. The attributes of the vehicles include their lengths, their desired and current speeds, their rates of acceleration, their positions along roadways, their positions in queues of vehicles, their destinations, the number of stops they make, their distances travelled, and the delay times they incur. The attributes of the traffic signals include the current phase of the signal cycle, as well as the remaining and predicted green times of future phases. The attributes of the roadways include the number of vehicles travelling along them as well as queue lengths of stationary vehicles along them.

The exogenous events of the model include the arrivals of new vehicles to the traffic network, while the endogenous events include the commencement of a vehicle's acceleration, deceleration or changing of lanes, as well as a phase change in the cycle of the traffic signals [27]. The service that the vehicles compete for is the green signal provided by the traffic signals, and all vehicles experience a delay if they are required to wait for service.

The framework was implemented in the Java-based simulation suite *Anylogic 6.8* [101]. AnyLogic is not a dedicated traffic simulation software package, but it does contain a *road traffic library*. This road traffic library allows for the modelling, simulation and visualisation of vehicle traffic in a network [102]. The library supports a detailed, yet highly efficient physical level of modelling of vehicle movement, making it suitable for modelling highway traffic, urban street traffic, on-site transportation at a manufacturing site, parking lots and many other systems comprising vehicles, roads and lanes. Furthermore, the library may be used to model very large-scale traffic systems as it is able to represent certain model parts at higher levels of abstraction using discrete event or system dynamics methods that are computationally less complex [102].

It is acknowledged that there are numerous commercially available microscopic traffic simulation software packages available (as was mentioned in §3.4), but the inner-workings of the majority of these packages are "black box" in nature, and do not allow the user to access or alter the fundamental settings of the embedded traffic signal control strategies. Thus, they were not suited to the scope and objectives of this study. Other commercially available microscopic traffic simulation software packages allow for the traffic signal timings to be altered, but not as functions of the input data provided by the assumed radar detection devices. AnyLogic was therefore chosen as it afforded the desired level of modelling flexibility. It provided a platform that is sufficiently customisable so as to allow for the extraction of necessary input data required by the various traffic signal control algorithms, as well as allowing for custom performance measure indicators to be provided as output.

The framework accommodates real-world data, as recorded by radar detection equipment (described in §1.2), as input values and provides for the accessibility of these data to a variety of traffic control algorithms which may be incorporated into the framework. In particular, dynamic, individual vehicle-specific data (such as vehicle lengths, positions along roadways, speeds, and rates of acceleration) are explicitly accommodated. Due to the modelling complexity associated with these characteristics, they are often omitted from traffic simulation models in favour of the adoption of simplifying assumptions (*e.g.* that all vehicles are of uniform length and travel at constant, uniform speeds). In an attempt to better gauge the proficiency of the various traffic control algorithms at reducing vehicle delay and facilitating coordination among intersections in as realistic a manner as possible, the framework also allows for the incorporation of user-specified performance measure indicators into the model which make use of individual vehicle data and characteristics — this feature is usually not available in commercial traffic simulation packages.

4.1.1 Building the road network and traffic signals

In the framework described above, the road network topology comprises a number of connected lines and arcs which represent the road segments on which vehicles travel. Before building the road network, it is required that certain global parameters are defined which dictate the appearance and connectivity of the road network. These parameters are the scale of the road network, the connection tolerance and the lane widths.

The scale defines the number of pixels per metre, thereby linking the unitless display of the modelling framework graphic with an actual unit of length. This is important when deciding how large the road network is going to be. For smaller road networks it makes sense to use a larger scale as it is physically easier to work with larger road shapes. For very large road networks, however, a smaller scale may be required in order to fit in the entire road network into the graphic display.

The connection tolerance (measured in pixels) is the maximum distance between two lane ends for which the two lanes are considered to be connected, *i.e.* if two lane ends are closer than the connection tolerance and form an obtuse angle, they are considered as connected, and a vehicle that exits one lane may continue travel on the other.

The lane width (measured in metres) defines how wide each lane in the road network will be, and as a result, how many lanes each road segment will contain. For example, if a line has a width of 60 pixels, the scale of the road network is 10 pixels per metre, and the lane width is set to 3 metres, then the corresponding road segment will comprise two lanes.

The default speed limit for the road network is also user-defined and is measured in metres per second.

4.1.2 The traffic control signals

The traffic signals positioned at each intersection in the model are modelled as individual agents and potentially operate independently of one another in order to facilitate the use of decentralised, self-organising traffic signal control strategies. The signal switching logic is controlled by means of a state chart which comprises various states and state transitions. An example of such a state chart is depicted in Figure 4.1. The number of different states in the state chart is determined by the number of phases which comprise a complete signal cycle at the intersection. A state transition occurs upon receipt of a specific message string issued by the traffic signal control algorithm implemented at the intersection. The frequency with which these messages are received depends on the traffic signal control implemented.

In Figure 4.1, each block, or state of the state chart represents a unique phase in the traffic signal cycle. The four distinct possible green phases of the cycle are depicted in Figure 4.2. The *HGreen* phase in Figure 4.1 corresponds to the display of green signals to all vehicles travelling from west to east and from east to west (see Figure 4.2(a)). During this phase, vehicles intending to turn right do so on a *permitted* basis (*i.e.* the vehicle turning right waits in the intersection until the intersection and a portion of road, which extends a user-specified distance on the opposite side of the intersection, are free of any on-coming traffic, at which point the vehicle completes its turn). Following this green phase, either the phase *HAllAmber* or the phase *HAmber is implemented.* During HAllAmber, an amber signal is displayed to all vehicles travelling from west to east and east to west, while during HAmber, amber signals are only displayed to vehicles travelling straight through the intersection or turning left. This is the start of an exclusive right-turn phase for vehicles travelling from west to east and east to west. The *Hred* phase is depicted in

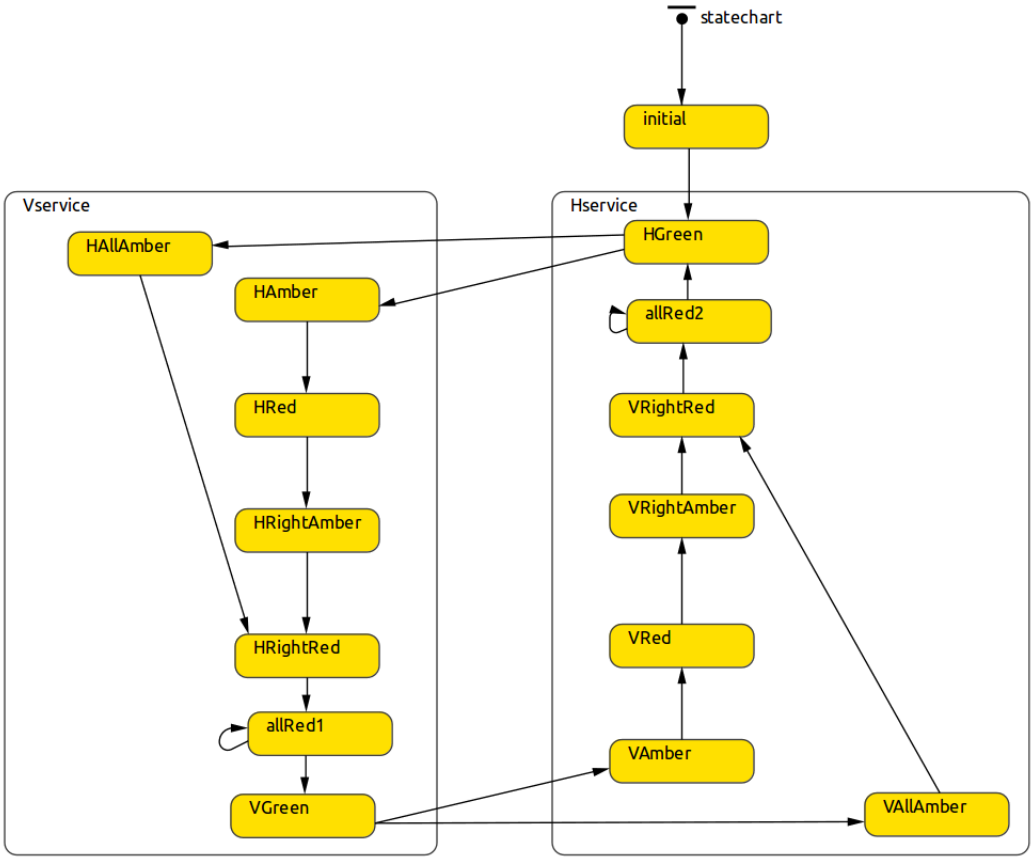


FIGURE 4.1: *An example of a state chart responsible for the phase transitions of traffic signals at an intersection.*

Figure 4.2(b). A green signal is only displayed to vehicles turning right, travelling from west to east and east to west. This is known as a *protected* or *exclusive* right-turn phase [94]. During the phase *HRightAmber*, an amber signal is displayed to all right-turning vehicles that were previously receiving a green signal, while red signals are displayed elsewhere. During the phases *HRightRed* and *allRed1*, all the signals of the intersection display red. A transition occurs from allRed1 to *VGreen* after a minimum allowable all-red period and as soon as the intersection is clear of all vehicles. The remaining phases follow the same order as those described above, except that they provide service to vehicles travelling in a north to south direction and south to north direction (see Figures 4.2(c)–(d)).

4.1.3 Populating the road network

With the road network and traffic signals in place, the next step is to introduce vehicles into the simulation model. Vehicles enter the road network at designated entry points. These vehicle arrivals may be defined according to one of four user-specified methods. The vehicles may arrive at a user-specified rate, in which case arrivals are stochastic and follow a Poisson distribution with a mean equal to the chosen rate. This is equivalent to specifying exponentially distributed interarrival times between vehicles with a mean equal to the inverse of the chosen rate. Alternatively, the user may specify a fixed inter-arrival time which would be identical for all arriving vehicles. The user may also choose to implement a stochastic rate schedule which defines how the arrival rate changes over time. Finally, the user may define a deterministic arrival schedule,

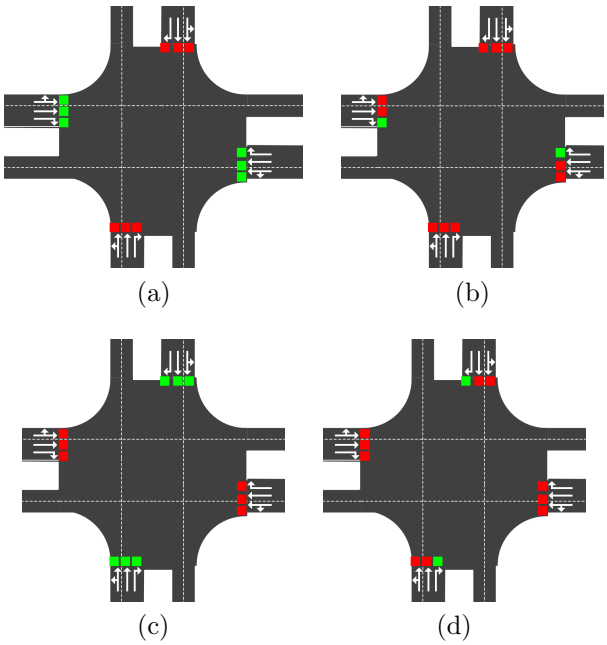


FIGURE 4.2: *Intersection traffic signal phase configurations. (a) All vehicles travelling from west to east or from east to west receive a green signal. Vehicles turning right do so on a permitted basis. (b) Exclusive right-turn phase for vehicles travelling from west to east or from east to west. (c) All vehicles travelling from north to south or from south to north receive a green signal. Vehicles turning right do so on a permitted basis. (d) Exclusive right-turn phase for vehicles travelling from north to south or from South to North.*

in which case the arrivals of vehicles are generated according to the exact times defined in the arrival schedule.

When a vehicle is generated in the simulation, several vehicle-specific parameters are defined instantaneously. These include the origin-destination pairing of the vehicle, the size of the vehicle, the vehicle's rates of acceleration and deceleration, and the vehicle's desired speed of travel. Vehicles are generated at each entry point to the road network, and upon generation the final destination of the vehicle is determined by Monte Carlo simulation. This origin-destination pairing of the vehicle dictates when and where a vehicle must change lanes, as well as whether it should turn left or right at an intersection, or carry on travelling straight. Monte Carlo simulation is used to determine the size of the vehicle generated. The user decides on the probabilities associated with the different sizes of vehicles which ultimately determines the number of small, medium and large vehicles present in the road network. A vehicle's size determines its rates of acceleration, deceleration and desired speed. Monte Carlo simulation is again employed to determine the desired speed of a vehicle, which is calculated by multiplying the general speed limit of the road network by a speeding factor drawn from a uniform, random distribution, the interval of which is user-specified. This distribution depends on the size of the vehicle. Typically, the larger the vehicle, the slower its rates of acceleration and deceleration, and the lower its desired travel speed. These trends may, however, be overridden by the user.

Apart from the simulation logic which determines how fast a vehicle travels, or at what rate it accelerates or decelerates, logic has also been implemented which determines when and where a vehicle must accelerate or decelerate. Associated with each vehicle are minimum and maximum allowable distances to the vehicle in front of it, which depend on the vehicle's speed, as well as minimum and maximum allowable speeds, which, in turn, depend on the distance to the vehicle in front of it. There is also a maximum allowable speed on curved roads (*e.g.* corners). Let v_i be

the speed of vehicle i and let $s_{i,i-1}$ be the distance between vehicle i and vehicle $i-1$ in front of it. Now, if $s_{i,i-1}$ is less than the value of some function f of v_i, which determines the minimum allowable distance between two vehicles, or if v_i is greater than the value of some function g of $s_{i,i-1}$, which determines the maximum allowable speed of a following vehicle, then vehicle i will decelerate. On the other hand, if $s_{i,i-1}$ is greater than the value of some function f' of v_i, which determines the maximum allowable distance between two vehicles or if v_i is less than the value of some function g' of $s_{i,i-1}$ which determines the minimum allowable speed of a following vehicle, then vehicle i will accelerate. The maximum speed on curved roads is determined according to a function h of the radius of the arc of the curve. The functions f, f', g, g' and h are all user-defined. The framework also provides an indication of when a vehicle is travelling at its desired speed, and when it is not. When a vehicle is travelling at its desired speed it is coloured blue; otherwise it is coloured white.

Explicitly incorporating individual vehicle acceleration into a traffic simulation model is not a trivial task, and typically results in a considerable increase in model complexity [13]. It is not uncommon, therefore, for traffic simulation models in the literature to omit these vehicle accelerations, assuming rather that vehicles are either stationary or travelling at their desired speeds. In such instances, alternative implicit techniques are employed to compensate for the delay times associated with finite vehicle accelerations. In [13], it was shown, however, that discrepancies resulted between the two approaches in terms of vehicle delay times experienced and vehicle queue lengths observed for a traffic network of signalised intersections as a direct result of this kind of omission. These discrepancies increased in magnitude as the number of intersections in the network increased as well as when the number of vehicles in the network increased. For this reason, and to add to the realism of the framework, it was decided that vehicle accelerations should be incorporated explicitly for the purpose of the study in this dissertation.

The logic responsible for a vehicle's interaction with traffic signals operates in much the same manner as that responsible for its interaction with other vehicles. When a red or a late amber signal is displayed, vehicles decelerate as if there were a stationary vehicle at the stop line of the intersection. In the case of permitted right-turning vehicles, while a green signal is displayed to oncoming traffic, the vehicle turning right will wait in the intersection until the intersection and a portion of road, which extends a user-specified distance on the opposite side of the intersection, are free of any on-coming traffic, at which point the vehicle will complete its turn. For the case in which a right-turning vehicle is still present in the intersection when the traffic signal changes from green to amber, the vehicle need only wait until the intersection is free of any oncoming traffic before completing its turn.

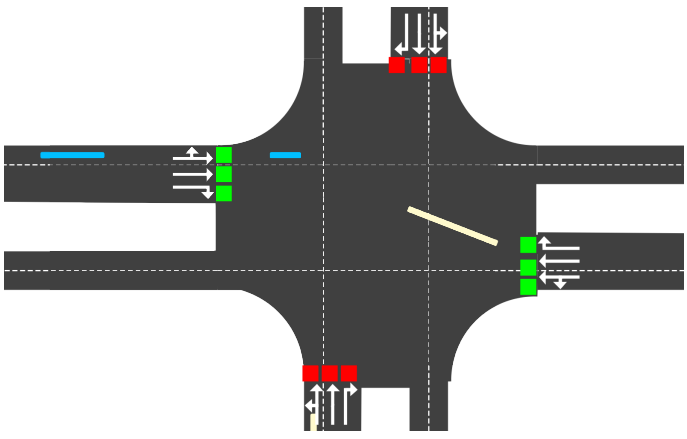


FIGURE 4.3: *An example of a permissive right-turn situation. The large white vehicle waits in the intersection until the two smaller vehicles travelling in a west-to-east direction have cleared the intersection.*

4.1.4 Data collection and assimilation

The traffic simulation modelling framework described in this chapter was designed to allow for the testing of traffic control algorithms which assume the use of the radar detection technology alluded to in §1.2 for the provision of all necessary input data. As was mentioned in §1.2, the radar detection sensors are capable of detecting and tracking individual vehicles in terms of their speeds, lengths, and distances to the intersections they are approaching. AnyLogic is capable of accurately reproducing these detection and tracking capabilities. In AnyLogic, it is possible to access an individual vehicle and obtain information pertaining to its characteristics (such as its length, speed and distance to the intersection) by calling applicable functions which are part of the *application programming interface* (API) of the road traffic library.

A challenge to this method, however, is that AnyLogic does not categorise the vehicles into sorted lists of any kind. Instead, all vehicles in the network are contained in one list. Thus, although it is possible to access the vehicles and their associated characteristics, it is a considerable challenge attempting to ascertain which vehicle is indeed being accessed, or how to access a specific vehicle. To overcome this challenge a list $\mathfrak{C}_j(t)$ was introduced for each approach lane j of each road section in the network. This list contains all vehicles present on approach lane j at time t. As a vehicle enters the lane, be it at the lane's entry point or as a result of a lane change, it is added to this list. A vehicle is removed from the list when it reaches the end of the lane or when it changes onto an adjacent lane. This list provides the user with information on the number of vehicles present along a specific lane. This makes it considerably easier to access specific vehicles and their associated characteristics, and is computationally more efficient than repeatedly searching through the entire list of vehicles in the network.

More information is provided in Chapter 6 of this dissertation on how the various traffic signal control algorithms utilise these data provided by the radar detection sensors.

4.2 Model output

Due to the fact that the desired speed, as well as the origin-destination pairing of a vehicle is known upon its generation, the total distance the vehicle has to travel and the delay time it experiences while travelling through the road network may be calculated. This method of calculation involves subtracting the time it would take the vehicle to move from its origin to its destination without being impeded by any traffic signals (and resulting queues) or slower moving vehicles (known as the vehicle's *ideal travel time*) from the actual time it spends travelling through the road network. The minimum time a vehicle can spend travelling through the road network is calculated by dividing the distance the vehicle has to travel from its origin to its destination by its desired speed. The actual time spent by a vehicle travelling through the road network is captured by a timing mechanism which records the time the vehicle enters the road network as well as the time it leaves the road network. Furthermore, the framework records the normalised delay time of vehicles in the network. This is considered to be an important measurement because not all vehicles travel the same distance through the network and therefore vehicles do not all experience the same potential for delay. This normalised delay time is calculated by dividing the actual time a vehicle spent in the network by the its ideal travel time. This normalised delay time value provides an indication of how long a vehicle spent travelling through the network in relation to its ideal travel time. For example, if a vehicle experiences a normalised delay time of 1.5, it indicates that the actual time taken for the vehicle to travel from its origin to destination took 0.5 times longer than its ideal travel time.

A novel performance measure additionally incorporated in the framework is the number of stops made by vehicles while travelling through the network. Associated with each vehicle is a variable representing the number of times the vehicle has come to a complete stop while in the network. This variable is incremented each time the vehicle comes to a complete stop and is recorded upon the vehicle's departure from the network. Just as all vehicles do not travel the same distance through the network and therefore do not experience the same potential for delay, so too all vehicles do not travel through the same number of intersections, and therefore do not experience the same potential for stopping. The normalised number of stops made by each vehicle is therefore recorded in the framework. The normalised number of stops made by a vehicle is calculated by dividing the number of stops made by the vehicle by the number of intersections it passes through while travelling from its origin to its destination. This value provides an indication of how often the vehicle was required to stop in relation to how many times it may be expected to stop. For example, if a vehicle is associated with a normalised number of stops value of 0.5, this indicates that the vehicle was only required to stop at half of the intersections it encountered.

The framework is also capable of measuring and recording the level of traffic saturation on the roadways of the network. Since the lengths of the roadways remain fixed, it is possible to calculate how heavily saturated the roadways are at any point in time by dividing the effective space occupied by vehicles in the network by the total combined lengths of all the roadways in the network.

Due to the stochastic nature of vehicle arrivals at an intersection, vehicle actuated traffic signal control algorithms typically implement differing green time durations from phase to phase. The framework is also able to capture the green time durations of the various phases for each signalised intersection in the traffic network.

For each of the four output values described above, the framework provides a comprehensive statistical analysis. This includes minimum, mean and maximum values, the mean confidence interval (assuming a confidence level of 95%), standard deviations, observation counts and the sum of all the observations recorded. An example of this output may be seen in Figure 4.4. In the figure, data are provided pertaining to the delay time experienced by vehicles in the traffic network so far. These data are updated in real time as the model runs. In addition to the count, mean, mean confidence interval, maximum, minimum, and standard deviation of the data, the output also bins the data and provides both a probability density function and a cumulative density function of the observations which enables histograms to be created from it. Furthermore, the framework allows for real-time analysis of the system to take place as all output generated may be displayed while the model is running. An example of this output is shown in Figure 4.5. Finally, the framework provides a breakdown of the above output for each direction of travel (for example, the average delay time experienced by all vehicles travelling from west to east).

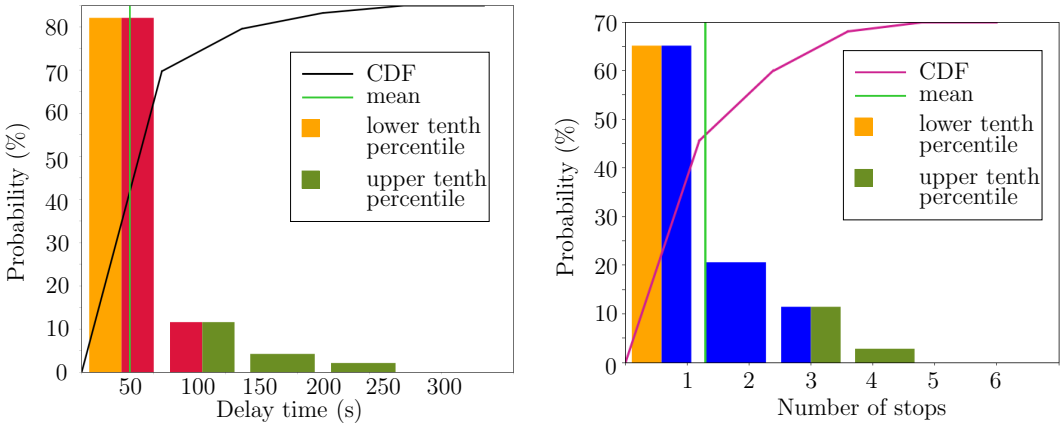
4.3 Model verification and validation

The verification and validation of a simulation model serve to fulfil two main objectives [3]. The first objective is to confirm that a model has been produced which represents true real-world system behaviour closely enough for the model to be used as a substitute for the actual real-world system for the purpose of experimentation. The second objective is to increase the credibility of the model to an acceptable level so as to enable the model to be used in any decision-making processes pertaining to the real-world system.

DelayTime

Count	529		
Mean	42.987		
Min	0		
Max	122.615		
Deviation	23.525		
Mean confidence	2.005		
Sum	22,740.058		
From	To	PDF(hits)	CDF(cum hits)
0	21	101	101
21	58.6	305	406
58.6	96.2	109	515
96.2	133.8	14	529
133.8	171.4	0	529

FIGURE 4.4: *An example of statistical data provided as output by the framework. In this particular instance, the delay times incurred by 529 individual vehicles have been recorded, as indicated by* Count. *The minimum of these delay times is zero seconds, while the maximum delay time incurred is 122.615 seconds, as indicated by* Min *and* Max, *respectively. The average delay time experienced is 42.987 seconds, as indicated by* Mean. *A confidence interval of 2.005 is associated with this mean, calculated at a 95% level of confidence, indicated by* Mean confidence. *Thus, with 95% confidence the mean is expected to lie between its current value plus or minus half of the mean confidence value. The standard deviation of the observations is equal to 23.525 seconds, as indicated by* Deviation. *The total delay experienced by all 529 vehicles is 22 740.058 seconds, as indicated by* Sum. *The associated probability density function (PDF) and cumulative density function (CDF) bins and frequencies may be seen in the lower half of the figure. The first row shows that 101 vehicles have incurred a delay time between zero seconds and 21 seconds.*



(a) *Probability density function and cumulative distribution function of the delay time experienced by vehicles.*

(b) *Probability density function and cumulative distribution function of the number of stops made by vehicles.*

FIGURE 4.5: *An example of dynamic output generated by a model in the framework described in this section while it runs.*

According to Banks [3], verification is concerned with building the model correctly and is utilised in the comparison of the conceptual model to the computer representation that implements that conception. The process of verification asks the following questions: *Is the model implemented correctly in the computer? Are the input parameters and logical structure of the model correctly*

represented? Similarly, Law [55] considers the verification of a simulation model to be concerned with determining whether the assumptions regarding the logic in a simulated representation of a real-world system have been translated correctly into a computer program, *i.e.* the successful debugging of a simulation model such that is free of any errors of logic.

Whereas verification is concerned with building the model correctly, validation is concerned with building the correct model. Validation is the process of confirming that a model is an accurate representation of the real-world system and is typically achieved through iterative calibration of the model. This process of calibration involves comparing the model to the real-world system and using discrepancies between the two, as well as the insights gained from these discrepancies, to improve the model [3]. This process is repeated until the desired level of model accuracy has been achieved. The iterative process of verification, validation and calibration is depicted in Figure 4.6.

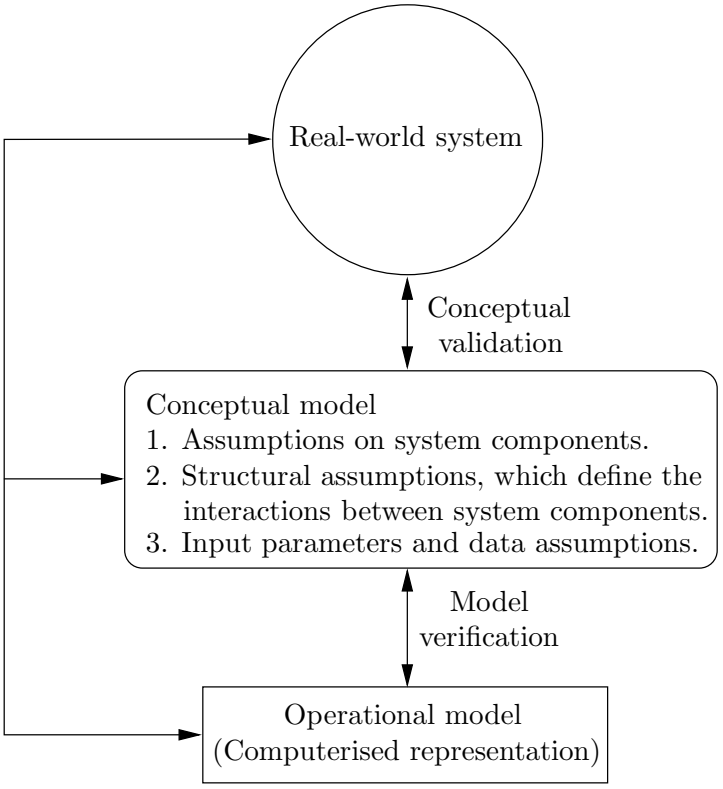


FIGURE 4.6: *The verification and validation process of a simulation model.*

4.3.1 Verification of the microscopic traffic simulation model framework

As is suggested by Law [55], the implementation of a model in a dedicated simulation software package greatly aids the verification process as it typically reduces the amount of coding required, thereby reducing the potential for coding errors. AnyLogic is an example of such a dedicated simulation software package, and offers a user-friendly interface as well as numerous well-documented predefined functions and API.

Animation is considered a very useful verification tool [5, 55]. In the building of this framework, the animation feature of AnyLogic was employed to visually detect any irregularities in the behaviour of system. Whenever any such illogical actions were detected, the code responsible could be identified and the error could be rectified.

Banks [5] recommends the use of an *interactive run controller* (IRC) or debugger to check that the model operates as intended. In general, the debugging of the programming code underlying a simulation model is an essential step in the verification process, and regular debugging is considered to be good programming practice and an important aspect of successful simulation model building, as outlined in §3.2. Implemented within AnyLogic is a pre-existing IRC and debugger. Before a model may be run in AnyLogic, its underlying code must first be compiled. Upon compilation of a model implemented in the framework described in this chapter, the debugger scans the model code for discrepancies and syntax errors. If found, attention is focused on the type of error and its location within the code, and possible suggestions for its rectification are provided. Once the model code has been found to be free of any errors following its compilation, the model may be run. In the event that an error is detected by the IRC during the running of the model, the model is suspended and an error report is generated. In this error report the nature and location of the error are described. In the building of the framework described in this chapter, a step-wise approach was employed whereby following any change or addition to the framework, model instances were compiled and run extensively, rectifying any errors if necessary before proceeding to the following step. This process involved running models in real time (*i.e.* 1 second of simulated time was equal to 1 second of real time) and in virtual time (*i.e.* at the maximum possible speed at which the simulation could be run). Running a model in real time allows for the visual inspection of the correctness of the new code addition or alteration. If necessary, a model can be run at a speed slower than real time too. Models were run in virtual time to verify that the addition or alteration of the code does not result in any errors over an extended period of time.

The *trace* feature of AnyLogic further aided in the verification process of the simulation framework. It is suggested in [3] that, during the verification process, simulation model output be closely examined for reasonableness under a variety of different input parameter settings by having the model print out comprehensive model statistics. Using the trace feature, the state of the simulation model may be displayed (or printed) at any point in time for any chosen event. For example, in the case of building this simulation model framework, the trace was used to print data pertaining to individual vehicle lengths, current speeds and rates of acceleration as well as distances to other vehicles or intersections for a number of model instances. Thus, while the visual animation capabilities of AnyLogic allow one to determine from a more abstracted point of view that the components of the framework are behaving in a logical manner (*e.g.* vehicles accelerating and decelerating when they are supposed to), the trace allows one to verify with precision that model logic is implemented accurately (*e.g.* the rate at which vehicles accelerate and their distances from the objects which resulted in the acceleration, be it another vehicle or an intersection).

4.3.2 Validation of the microscopic traffic simulation model framework

Much like verification, the validation of the microscopic traffic simulation modelling framework described in this chapter was performed throughout its inception and building. A subjective technique described in [5] in respect of the validation of a simulation model is sensitivity analysis. This technique validates the *continuity* of the model in question. For example, in the case of the framework described in this chapter, the average arrival rate of vehicles to the system would increase while the traffic signal phase timings remain fixed. This resulted in an increase in average vehicle delay time and increased queue lengths, as one would expect in real-world conditions. Banks [5] goes on to suggest performing so called *extreme-condition tests* in which input data are set to extreme values. Increasing the vehicle arrival rates to far greater than the maximum flow rate resulted in saturation of the traffic network, as expected.

In addition to validating the continuity of the model, the *consistency* of the model was also validated. This was achieved by performing multiple simulation runs in which the input parameters remained the same for each simulation run, ensuring that the results of each simulation run are comparatively *similar*. This was indeed observed to be the case for the modelling framework described in this chapter. The results for each simulation run were similar, but not identical, for despite the input parameters remaining unchanged for each simulation run, the inter-arrival times of the vehicles as well as the variables which determine their sizes, speeds and destinations are generated stochastically. It was found that for each set of input parameters, the models implemented in the framework produced consistently similar results in terms of the various output statistics, such as the number of vehicles passing through the traffic network, their average and maximum delay times, the average and maximum number of stops made by vehicles, and the average saturation levels of the traffic network.

The third step towards validation of the framework, was comparing the output generated by a model built within it, with that of a real-world system. The data used for this comparison were compiled into a collection of traffic counts for the intersection of Adam Tas Road and Bird Street in Stellenbosch. A satellite image of the intersection may be seen in Figure 4.7. The data

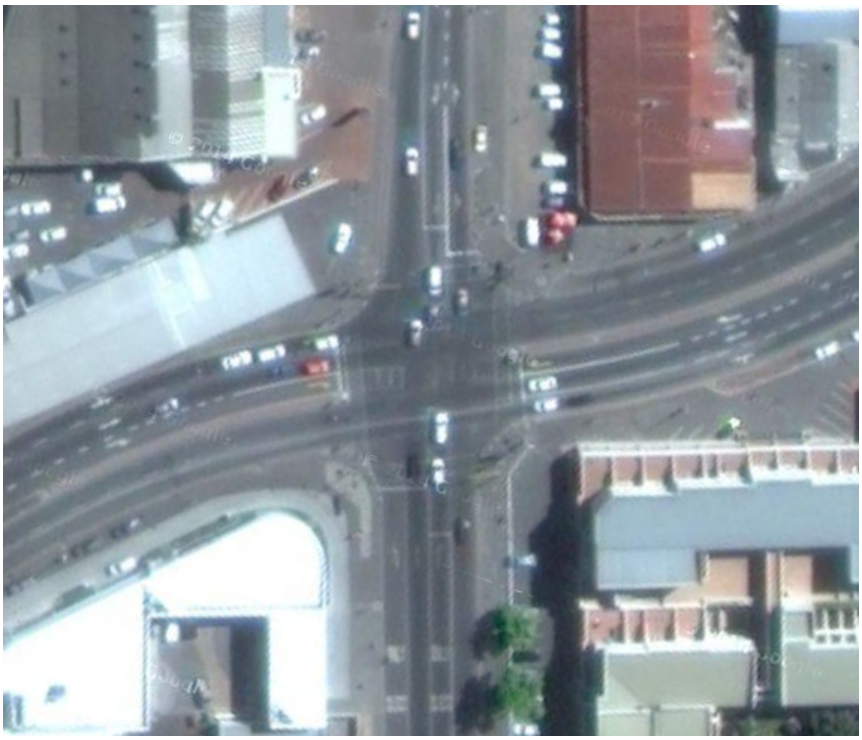


FIGURE 4.7: *A satellite image of the Adam Tas Road and Bird Street intersection in Stellenbosch. In the image, Adam Tas Road may be seen to run in a west to east and east to west direction, while Bird Street may be seen to run in a north to south and south to north direction. Each approach comprises three lanes. Vehicles travelling along the left-most lane of each approach may travel straight or turn left at the intersection, vehicles travelling along the middle lane may only travel straight through the intersection, while the right-most lane is an exclusive right-hand turning lane.*

were collected by the *Stellenbosch Traffic Department* for a previous study performed on the intersection by Van der Merwe [95]. These data are presented in Appendix A. The data were collected in fifteen minute intervals between 06:30 and 18:00 on a Tuesday during the university and school term so as to accurately represent normal traffic conditions. The data comprised the number of vehicles that travelled straight through the intersection, turned left at the intersection or turned right at the intersection for each of the four intersection approaches. In addition to the

vehicle counts, the signal timings of each green phase from several signal cycles were provided, for the morning-peak, midday, and afternoon-peak periods. These green phases are depicted in Figure 4.8.

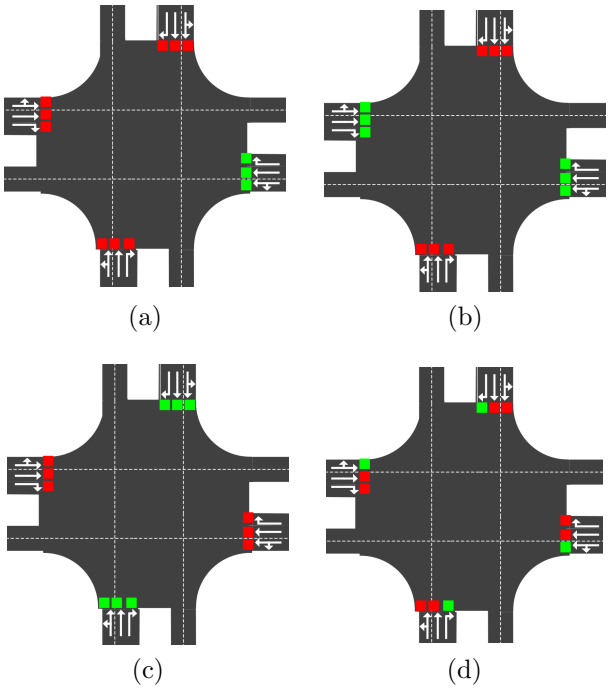


FIGURE 4.8: *The green phases implemented at the Adam Tas Road and Bird Street intersection in Stellenbosch. (a) All vehicles travelling along Adam Tas Road West (AT(W)) receive a green signal. (b) All vehicles travelling along Adam Tas Road West (AT(W)) and Adam Tas Road East (AT(E)) receive a green signal. Vehicles turning right do so on a permissive basis. (c) All vehicles travelling along Bird Street North (BS(N)) and Bird Street South (BS(S)) receive a green signal. Vehicles turning right do so on a permissive basis. (d) All vehicles turning right from Bird Street North (BS(N)) and travelling along Bird Street South (BS(S)) receive an exclusive turning phase as do vehicles turning left from Adam Tas Road West (AT(W)) and travelling along Adam Tas Road East (AT(E)).*

To validate the model, the vehicle throughput data of each intersection approach was aggregated into ten one hour-long periods and one 30 minute period. For these periods, the average vehicle arrival rate to the intersection along the approach was calculated, as well as the turning probabilities of the vehicles travelling along them. The individual signal timings of each phase were averaged for each of the morning-peak, midday and afternoon-peak periods and are summarised in Table 4.1.

Period	Traffic signal phase			
	Phase 1 (s)	Phase 2 (s)	Phase 3 (s)	Phase 4 (s)
Morning peak (06:30–09:30)	10.00	55.14	55.79	14.71
Midday (09:30–16:30)	12.38	46.75	68.88	9.75
Afternoon peak (16:30–18:00)	8.11	52.84	70.74	3.79

Table 4.1: *The average green signal phase timings at the Adam Tas Road and Bird Street intersection in Stellenbosch for the morning, midday and afternoon-peak periods.*

These aggregate values of vehicle arrival rates, turning probabilities and signal phase timings were used as input to the simulation model. The simulation model was run for the equivalent

of eleven and a half hours, with output being generated and recorded every hour pertaining to the number of vehicles arriving at the intersection along each approach, as well as the number which travel straight through the intersection or turn left or right. The model was run 30 times, with the output of each run being averaged and compared to the real-world observation to determine how accurately the model represents the system it is attempting to replicate. The results of these comparisons may be seen in Table 4.2. It may be seen from the results in the table that the model closely resembles the real-world system on which it is based. After eleven and a half hours of varying arrival rates, turning probabilities and signal timing settings, the average total number of vehicles recorded as passing through the intersection was only 0.5% less than the actual observed value. After eleven and a half hours of simulated traffic flow, the largest discrepancy between the model output and the collected data was that of the number of vehicles which turned right at the intersection from Adam Tas Road East (AT(E)), with the model estimation of the number of vehicles which turned right recorded as 4.9% below that of the collected data. These results give a clear indication that models built within the framework described in this chapter may be relied upon to provide valid and representative results.

4.4 Chapter summary

A microscopic traffic simulation framework which was built for the purpose of the study contained in this dissertation was described in this chapter. In §4.1 the various components of the framework were described, including the traffic signals, the vehicles which populate the road network as well as the roads on which they travel. In §4.2, the array of output data which may be generated by traffic simulation models implemented within this framework was described. The chapter closed in §4.3 with a discussion on the procedures that were followed to verify and validate the framework.

Approach	Collected data	Simulation output	Absolute % error	Approach	Collected data	Simulation output	Absolute % error
	Comparison after 1 hour				Comparison after 2 hours		
AT(W) L	255	256.4	0.6 %	AT(W) L	449	449.8	0.2 %
AT(W) S	503	490.5	2.5 %	AT(W) S	986	981.8	0.4 %
AT(W) R	7	4.5	35.2 %	AT(W) R	20	20.0	0.2 %
BS(S) L	5	5.7	14.0 %	BS(S) L	16	17.6	10.0 %
BS(S) S	216	220.4	2.0 %	BS(S) S	539	545.0	1.1 %
BS(S) R	99	91.8	7.3 %	BS(S) R	302	292.4	3.2 %
AT(E) L	195	198.5	1.8 %	AT(E) L	555	557.3	0.4 %
AT(E) S	625	623.0	0.3 %	AT(E) S	1286	1282.7	0.3 %
AT(E) R	128	118.0	7.8 %	AT(E) R	279	265.6	4.8 %
BS(N) L	31	31.1	0.4 %	BS(N) L	77	78.9	2.4 %
BS(N) S	447	436.4	2.4 %	BS(N) S	982	958.1	2.4 %
BS(N) R	347	333.7	3.8 %	BS(N) R	667	624.7	6.3 %
Total	**2858**	**2810.0**	**1.7%**	**Total**	**6158**	**6074.0**	**1.4 %**

Table 4.2: *An hourly comparison of the collected real-world data and the output of the simulated model attempting to replicate the real-world system. (Continued on next page).*

Approach	Collected data	Simulation output	Absolute % error
Comparison after 3 hours			
AT(W) L	672	671.9	0.0 %
AT(W) S	1363	1365.8	0.2 %
AT(W) R	37	35.8	3.3 %
BS(S) L	36	36.2	0.5 %
BS(S) S	825	838.2	1.6 %
BS(S) R	465	450.2	3.2 %
AT(E) L	762	771.8	1.3 %
AT(E) S	1643	1649.7	0.4 %
AT(E) R	367	352.4	4.0 %
BS(N) L	133	134.2	0.9 %
BS(N) S	1436	1445.7	0.7 %
BS(N) R	908	874.8	3.7 %
Total	**8647**	**8626.5**	**0.2 %**
Comparison after 4 hours			
AT(W) L	872	869.9	0.2 %
AT(W) S	1742	1739.3	0.2 %
AT(W) R	56	54.9	1.9 %
BS(S) L	46	47.6	3.6 %
BS(S) S	1136	1157.2	1.9 %
BS(S) R	626	605.1	3.3 %
AT(E) L	963	976.4	1.4 %
AT(E) S	1963	1974.6	0.6 %
AT(E) R	446	427.2	4.2 %
BS(N) L	187	188.9	1.0 %
BS(N) S	1829	1848.7	1.1 %
BS(N) R	1148	1115.2	2.9 %
Total	**11014**	**11005.0**	**0.1 %**
Comparison after 5 hours			
AT(W) L	1086	1087.6	0.1 %
AT(W) S	2152	2152.5	0.0 %
AT(W) R	68	67.1	1.3 %
BS(S) L	61	64.4	5.6 %
BS(S) S	1440	1458.2	1.3 %
BS(S) R	804	778.3	3.2 %
AT(E) L	1153	1169.2	1.4 %
AT(E) S	2285	2302.3	0.8 %
AT(E) R	532	509.4	4.2 %
BS(N) L	249	248.2	0.3 %
BS(N) S	2184	2209.1	1.1 %
BS(N) R	1366	1330.2	2.6 %
Total	**13380**	**13376.6**	**0.0 %**
Comparison after 6 hours			
AT(W) L	1298	1296.7	0.1 %
AT(W) S	2524	2525.8	0.1 %
AT(W) R	90	90.6	0.7 %
BS(S) L	75	76.5	2.0 %
BS(S) S	1767	1796.2	1.7 %
BS(S) R	985	957.0	2.8 %
AT(E) L	1351	1363.9	1.0 %
AT(E) S	2616	2636.3	0.8 %
AT(E) R	620	593.1	4.3 %
BS(N) L	327	326.6	0.1 %
BS(N) S	2514	2551.4	1.5 %
BS(N) R	1625	1576.4	3.0 %
Total	**15792**	**15790.5**	**0.0 %**
Comparison after 7 hours			
AT(W) L	1531	1524.5	0.4 %
AT(W) S	2938	2946.2	0.3 %
AT(W) R	111	111.7	0.6 %
BS(S) L	101	100.9	0.1 %
BS(S) S	2140	2176.4	1.7 %
BS(S) R	1199	1169.6	2.4 %
AT(E) L	1553	1566.6	0.9 %
AT(E) S	3008	3037.6	1.0 %
AT(E) R	703	670.4	4.6 %
BS(N) L	387	386.7	0.1 %
BS(N) S	2919	2959.8	1.4 %
BS(N) R	1893	1838.9	2.9 %
Total	**18483**	**18489.4**	**0.0 %**
Comparison after 8 hours			
AT(W) L	1751	1749.2	0.1 %
AT(W) S	3341	3353.2	0.4 %
AT(W) R	133	132.6	0.3 %
BS(S) L	120	118.5	1.3 %
BS(S) S	2508	2552.6	1.8 %
BS(S) R	1410	1374.0	2.6 %
AT(E) L	1742	1760.9	1.1 %
AT(E) S	3378	3419.9	1.2 %
AT(E) R	804	764.7	4.9 %
BS(N) L	412	411.2	0.2 %
BS(N) S	3176	3226.4	1.6 %
BS(N) R	2080	2023.3	2.7 %
Total	**20855**	**20886.3**	**0.2 %**
Comparison after 9 hours			
AT(W) L	1995	1995.7	0.0 %
AT(W) S	3805	3815.6	0.3 %
AT(W) R	146	147.1	0.8 %
BS(S) L	139	136.6	1.7 %
BS(S) S	2918	2965.4	1.6 %
BS(S) R	1629	1586.9	2.6 %
AT(E) L	1940	1956.7	0.9 %
AT(E) S	3803	3846.0	1.1 %
AT(E) R	901	858.7	4.7 %
BS(N) L	464	464.9	0.2 %
BS(N) S	3542	3596.1	1.5 %
BS(N) R	2296	2231.5	2.8 %
Total	**23578**	**23601.3**	**0.1 %**
Comparison after 11.5 hours			
AT(W) L	2659	2658.5	0.0 %
AT(W) S	5028	4868.6	3.2 %
AT(W) R	194	193.4	0.3 %
BS(S) L	173	171.7	0.7 %
BS(S) S	4202	4290.9	2.1 %
BS(S) R	2415	2321.8	3.9 %
AT(E) L	2559	2579.5	0.8 %
AT(E) S	4815	4867.4	1.1 %
AT(E) R	1176	1118.1	4.9 %
BS(N) L	593	584.4	1.5 %
BS(N) S	4736	4746.8	0.2 %
BS(N) R	2820	2827.5	0.3 %
Total	**31370**	**31228.6**	**0.5 %**

CHAPTER 5

Prevailing traffic signal control paradigms in the literature

Contents

Traditional traffic signal control strategies (*i.e.* both the classical fixed-time control strategies and the more recently developed vehicle actuated control strategies) as well as *self-organising* traffic signal control strategies are described in this chapter. In §5.1 examples of current state-of-the-art traffic control strategies are briefly discussed. Two examples of fixed-time traffic control strategies from the literature are described in §5.2, while in §5.3 two examples of self-organising traffic control strategies from the literature are described. The mechanisms and logic behind each strategy are considered, as are the associated advantages and limitations of these strategies. The traffic signal control strategies described in this chapter will form benchmarks against which the novel traffic signal control strategies proposed later in this dissertation will be tested.

5.1 Existing traffic signal control

In a classical fixed-time traffic signal control paradigm, constant predetermined parameters are implemented in a continually repeating cyclical manner. For an isolated intersection, these parameters include the cycle time and the duration of the various green times within this cycle [33, 68, 99]. Coordination among adjacent signalised intersections may be achieved by implementing the same cycle time at each intersection and adjusting the *offset* between them for displaying a green signal [34]. This offset is usually implemented so as to facilitate the formation and propagation of green waves along certain corridors. The problem of optimising these traffic signal control parameters for a given traffic network is, however, NP-hard [79]. For this reason, the

cycle length, green-time split and offsets among adjacent intersections are typically optimised off-line for certain standard scenarios (*e.g.* morning or afternoon rush-hours) and applied under the corresponding prevailing traffic conditions [53]. There are, however, several limitations associated with fixed-time traffic control. The key limitation is that the traffic signal cycle times, green-time splits and offsets are optimised for assumed average traffic scenarios which are rarely realised precisely and are therefore not, in fact, optimal for an actual real-time traffic situation [53]. This may lead to green times being longer than actually required, particularly during periods of low traffic demand (*e.g.* at night) and as a result incompatible traffic flows may experience unnecessarily long, yet avoidable delay times. Moreover, coordination of green signals among adjacent intersections is typically applicable to only one or two traffic corridors and flow directions, which tends to obstruct opposite and crossing traffic flows.

Advances have been made in respect of the classical cycle-based fixed-time traffic signal control strategy. These more advanced concepts attempt to adjust the traffic signal control parameters in response to slow or systematic variations of the average traffic demand [54]. Two such examples are the commercially available *Split Cycle Offset Optimisation Technique* (SCOOT) [50, 87] and *Sydney Coordinated Adaptive Traffic System* (SCATS) [62, 86]. SCOOT relies on data stored in a "SCOOT database" pertaining to the physical layout of the road network as well as data provided by upstream detectors (in the form of electromagnetic induction loops) to determine *cyclic flow profiles* [87]. Once a vehicle is detected, SCOOT models its progression from the detector to the intersection. These input data are utilised by three optimisers which continually adapt the split, offset and cycle times of all intersections in the SCOOT controlled area [87]. While SCATS also relies on input data provided by upstream loop detectors, it implements "logic and algorithms to analyse real-time traffic data to produce timings that are suitable for prevailing traffic conditions"[86]. Both SCOOT and SCATS are a form of centralised traffic signal control which are intended for use in conjunction with a broader traffic control system typically involving some form of human supervision. While more effective at reducing delay time than classical fixed-time control, these two examples of attempted online optimisation of traffic signal timings are essentially only variations on the theme of a mainly cyclic mechanism of control. Alternative heterogeneous phase plans, which vary the order and the frequency of service of the different intersection approaches, are not considered [54].

In an attempt to improve on the responsiveness of traffic control strategies to fluctuations in prevailing traffic conditions, various alternative approaches have been introduced in the literature for accommodating acyclic traffic control and permitting alternate orderings of green phases. Examples of these strategies include PRODYN [45], *Optimised Policies for Adaptive Control* (OPAC), *Urban Traffic Optimisation by Integrated Automation* (UTOPIA) [64], and *Adaptive Limited Lookahead Optimisation of Network Signals — Decentralised Version* (ALLONS-D) [83]. These approaches all implement a *rolling horizon*[1] optimisation technique [14, 54], but differ in their approaches to solving the combinatorial optimisation problem at hand. PRODYN optimises traffic signal timings by adopting a forward dynamic programming approach [14], while OPAC essentially enumerates the solution space [54], selecting signal phase sequences and timings based on associated performance indices which are calculated using dynamic programming techniques [58]. UTOPIA is a hybrid control system combining both online dynamic and offline optimisation [14] which comprises both a system-wide optimiser as well as local optimisers, and relies on the prediction of traffic flow through the network. ALLONS-D traverses a complex decision tree by means of a back-tracking algorithm [54, 83]. A limitation shared by all of these strategies is that they rely on vast amounts of advance data collection and processing, and re-

[1] A planning horizon is partitioned into a *head* period which contains actual detected traffic data and a *tail* period with assumed synthetic traffic data. An optimal policy is determined for the entire horizon, but implemented only for the head period. This process continues to repeat itself as new data become available [14].

quire considerable processing power [44]. Furthermore, for the case of global coordination in a traffic network, the data required (such as dynamic origin-destination matrices of vehicles as well as turning probabilities) are difficult to quantify and obtain [44].

5.2 Fixed-time traffic signal control

In the case of signalised traffic control at road intersections it is desirable to discover the most suitable cycle length for the intersection, as well the green time afforded each lane or group of lanes approaching the intersection. Many methods have been proposed for accomplishing this, notable examples being those found in the *Highway Capacity Manual* [94], published by the Transportation Research Board [2]. Before considering the methods and procedures proposed in the Highway Capacity Manual, a few key aspects of traffic control at signalised intersections are first defined.

In order to determine the best cycle length and green-time proportions for a given intersection, it is required that a *phase plan* and *signal type* be established for the intersection [94]. The *phase plan* of the intersection comprises the number of phases to be implemented at the intersection as well as the order or sequence in which they will be implemented. Generally, an intersection employs either two-phase control, or multi-phase control. Two-phase control is the simplest available phase plan in which each of two intersecting streets is afforded a green time during which all vehicle movements are permitted, with all right-turns occurring on a permissive basis (*i.e.* vehicles turning right wait for a suitably sized gap in on-coming traffic before turning). An example of a simple two-phase control plan with associated queue length evolutions is shown in Figure 5.1. Multi-phase control, on the other hand, is employed when the volume of vehicles turning is large enough to warrant a protected turning phase.

The *signal type* of an intersection may be one of three types: pre-timed, fully actuated or semi-actuated. The *pre-timed signal type* involves a fixed sequence of phases which are repeated continually, the green times of which are fixed, resulting in a constant cycle length. The *fully actuated signal type* employs some sort of vehicle detection on all approaches to the intersection which is used to determine the timings for each phase — as a result, the cycle lengths tend to vary from cycle to cycle. The *semi-actuated signal type* typically only employs vehicle detection on the minor approaches to the intersection, *i.e.* the approaches with lighter traffic flow. In addition to deciding on the phase plan and signal type to be implemented at an intersection, it is also necessary to acquire an understanding of the dynamics associated with the intersection.

5.2.1 A fixed-time traffic signal control approach from the literature

The initial steps of the approach in [94] are the determination of approach *capacities* and *volume to capacity ratios* together with the identification of *critical lane groups*. The definition of capacity at signalised intersections is based on the concept of maximum, or saturation, flow rates. The *capacity* c_j of approach j, is defined as the maximum number of vehicles that would pass through the intersection at the rate $Q_j^{\max}$ during the effective green time g_j received by approach j per cycle of duration C. It is given by the expression

$$c_j = Q_j^{\max} \frac{g_j}{C}, \tag{5.1}$$

[2] This publication is held in high esteem by the transportation research community and has for many years been a worldwide reference for transportation and traffic engineering scholars and practitioners, as well as the basis for several country-specific capacity manuals.

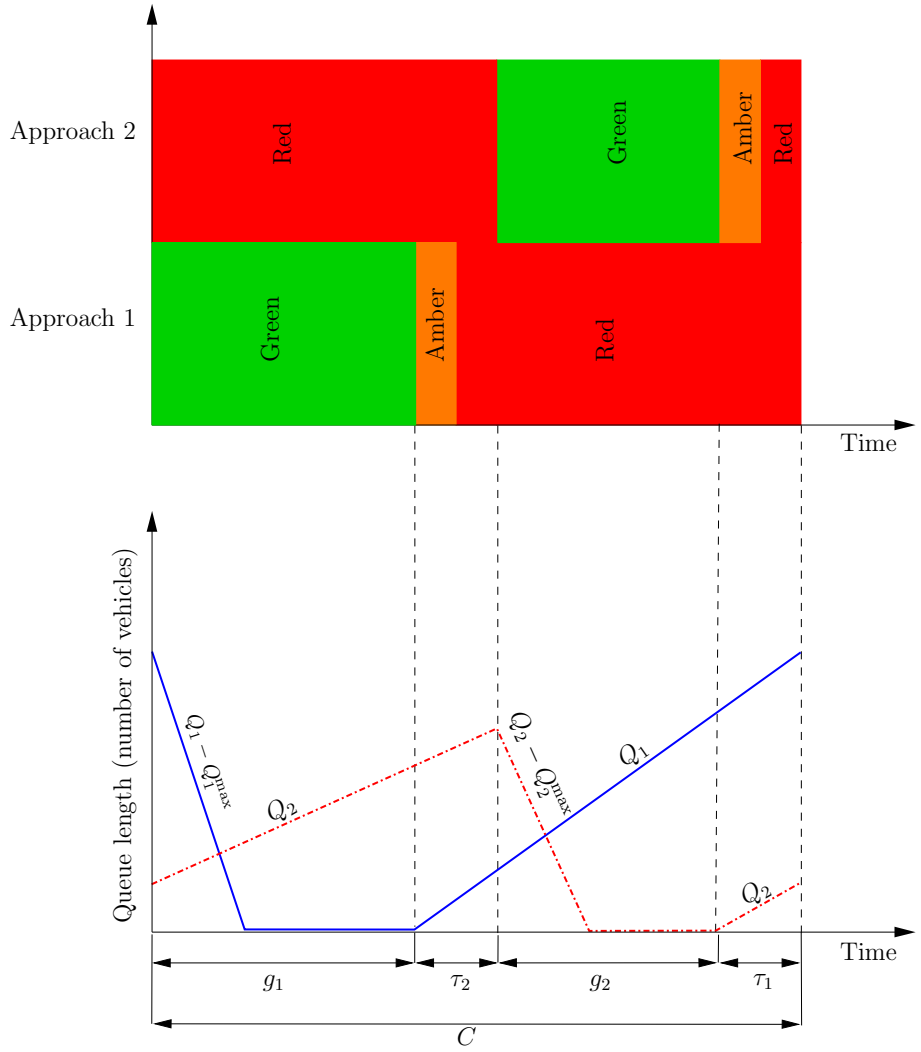


FIGURE 5.1: *An example of a two-phase fixed-time traffic control strategy and associated queue lengths.*

where g_j/C is the so-called *green ratio* for approach j.

The *flow rate to capacity ratio* of an approach, denoted by F_j, also known as the *volume to capacity ratio*, is a measure of the degree of saturation along approach j. It is given by

$$F_j = \frac{Q_j}{c_j} = \frac{Q_j}{Q_j^{\max}\frac{g_j}{C}} = \frac{Q_j C}{Q_j^{\max} g_i} \tag{5.2}$$

and may be interpreted as the total number of vehicles along approach j that require service at the intersection per cycle, divided by the maximum number of vehicles that may be served during the effective green time allocated to approach j.

The *critical lane group* of an intersection is a grouping of lanes or approaches achieving the highest volume flow rate to saturation flow rate ratio. For example, in a scenario in which at least two approach lanes are served during the same green phase, one of these approach lanes will typically require more green time than the rest (*i.e.* it will have a greater flow rate to capacity ratio). This approach lane is considered the critical lane for its associated green phase. Each green phase of a signal cycle is associated with a critical approach lane which determines the green time requirements of that phase. These critical approach lane values are used to determine

the critical volume to capacity ratio for the entire intersection, F_c, assuming that green times are allocated proportionally according to the volume flow rate to saturation flow rate ratios of the critical approach lanes. It is given by the sum of the flow rate to capacity ratios of all the critical approach lanes. The critical capacity of an intersection which assumes a fixed-time signal control approach with a signal cycle comprising m phases with corresponding critical approach lanes $j \in \{1, \ldots, m\}$ is given by

$$F_c = \sum_{j=1}^{m} \frac{Q_j C}{Q_j^{\max} g_j} = \sum_{j=1}^{m} \frac{Q_j}{Q_j^{\max}} \left(\frac{C}{C - \sum_{j=1}^{m} \tau_j} \right). \tag{5.3}$$

In (5.3), τ_j is the setup time associated with the green phase during which critical approach lane j is served, and $\sum_{j=1}^{m} \tau_j$ represents the total time per cycle during which vehicles may not enter the intersection. With the vehicle flow rates along all intersection approach lanes known it is possible to determine a cycle length for the computed flow ratios and desired critical flow ratio for the intersection, which is a user-defined input. This cycle length is estimated by

$$C = \frac{\sum_{j=1}^{m} \tau_j F_c}{F_c - \sum_{j}^{m} \frac{Q_j}{Q_j^{\max}}}. \tag{5.4}$$

The effective green time of critical approach lane j is given by

$$g_j = \frac{Q_j C}{Q_j^{\max} F_j} \tag{5.5}$$

and the minimum cycle length, which will avoid saturation along any of the critical approaches, is

$$C_{\min} = \frac{\sum_{j=1}^{m} \tau_j}{1 - \sum_{j}^{m} \frac{Q_j}{Q_j^{\max}}}, \tag{5.6}$$

i.e. the value of C in (5.4) where $F_c = 1$.

5.2.2 The green-wave method

A method is proposed in [38] to facilitate the coordination between intersections which implement a fixed-time traffic signal control approach. This so-called *green-wave method* or *coordinated signal control method* implements an offset between the switching of traffic signals of consecutive intersections. This offset is equal to the expected vehicle travel time between the intersections. This method is particularly effective when the majority of traffic flows in the direction of the green waves at low densities. A disadvantage of the method is that it facilitates the propagation of green waves in only two directions at best. Thus, if a vehicle is travelling in an opposite direction to the green wave, it will typically encounter numerous red traffic signals and, therefore, experience a greater delay. Furthermore, if vehicles travel at speeds faster or slower than anticipated, then their arrivals at intersections may be out of sync with the green signals, and will therefore be delayed.

To implement the green wave method, a fixed time period T is selected such that T is a multiple of the distance from one intersection to the next (in [38] it is assumed that intersections are uniformly spaced throughout the network and that vehicles traverse one unit of distance per time unit). Associated with each intersection I is an offset value b_I which is determined according to

the displacement of the intersection from the centre of the traffic network. An expression for b_I is given by

$$b_I = \left\lfloor (I_x - I_y) \bmod \frac{T}{2} + 0.5 \right\rfloor , \tag{5.7}$$

where I_x and I_y are the displacements of intersection I from the centre of the traffic network along the x-axis and y-axis, respectively. The offset value in (5.7) results in green waves which move from a west-to-east direction and a north-to-south direction. The traffic signals at intersection I are switched exactly twice every T time units, whenever a global counter (which cycles from zero to $\frac{T}{2}$) is equal to b_I.

5.3 Self-organised traffic signal control

Decentralised, self-organising traffic control has been proposed as an attractive alternative to solving the complex combinatorial optimisation problem of signalised traffic intersection control by either enumerating large solution spaces or traversing complex decision trees of possible switching sequences. The self-organising traffic control algorithm proposed by Gershenson [38], as well as that proposed by Lämmer and Helbing [53], are briefly reviewed in this section.

5.3.1 Gershenson's traffic signal control algorithm

Gershenson [37] claims that improving vehicle flow at signalised intersections and obtaining coordination among adjacent intersections is more a problem of adaptation rather than optimisation. The self-organising traffic control algorithm proposed by Gershenson and Rosenblueth [38] comprises six rules which are implemented independently at each signalised intersection to regulate traffic, with no direct communication among intersections. The model assumes that all vehicles are of the same size and all travel at an equal, uniform speed. In [38], the algorithm is implemented at an abstract level using elementary cellular automata[3] [85]. Time is assumed to be discrete, and all cells are updated synchronously. A further assumption is that traffic may travel in one direction along a roadway only and that no turning is allowed at intersections.

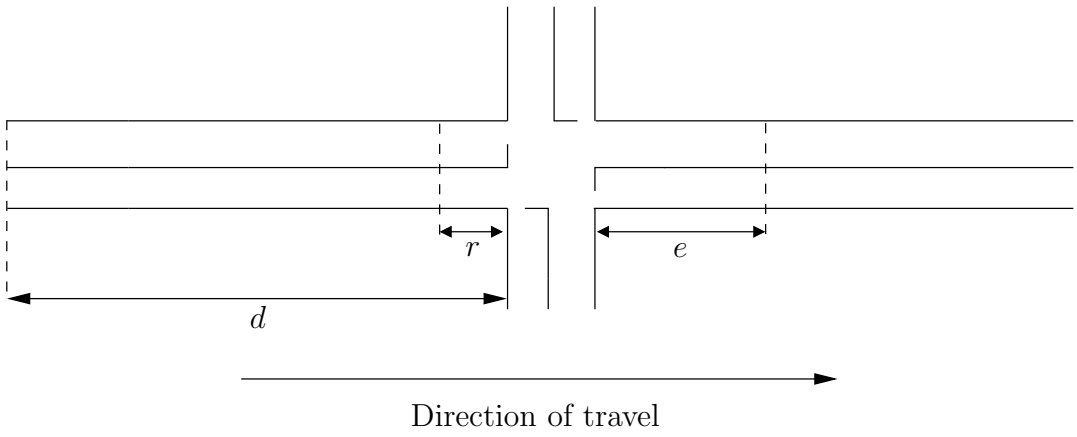


FIGURE 5.2: *An example of an intersection illustrating the distances d, r and e which are central to the self-organising algorithm proposed by Gershenson and Rosenblueth in [38].*

[3] An elementary cellular automaton is a collection of cells arranged on a one-dimensional array. A cell may occupy one of two possible states, 0 if there is no vehicle present in it, or 1 if there is a vehicle present in it [38].

The first rule states that for every time step a counter φ_I is incremented by the number of vehicles waiting behind or approaching a red signal within a specified distance d of intersection I. The signals are switched (from red to green and green to red) when the value of this counter exceeds a pre-specified threshold value ς_I, at which time the counter is reset to zero. This rule ensures that service is provided to an approach as soon as the amount of accumulated time spent by vehicles within a distance d of a red signal exceeds ς_I. It is thus intended that if a large platoon of moving vehicles were to be detected upstream, they would trigger a signal change and be served in a green wave manner. This rule also facilitates the formation of platoons by requiring vehicles to wait behind a red signal until sufficiently many have arrived before a switch is triggered.

The second rule states that once a traffic signal switches to green, it must remain green for at least u_I time steps. This rule prevents excessive switching of the traffic signals during high traffic demand scenarios. A second counter $t_{I_{\text{green}}}$ keeps track of how long a green signal has been displayed at intersection I. A signal change may only occur when $t_{I_{\text{green}}} > u$, at which point $t_{I_{\text{green}}}$ is reset to zero.

The third rule states that if there are at most $s > 0$ vehicles intending to cross the intersection, which are approaching a green signal and are within a distance r of the intersection, then the signal may not switch from green to red. This rule is designed to maintain the integrity of platoons by preventing the tail of the platoon from being separated. The rule does, however, allow for the separation of long platoons.

The fourth rule states that if no vehicle is approaching a green signal within a distance d of the intersection but at least one vehicle is approaching a red signal within a distance d from the intersection, then the signals and hence the provision of service must change. This rule facilitates the rapid switching of service during low traffic demand scenarios, allowing individual vehicles to trigger a switch in demand rather than requiring them to wait for a sufficiently large platoon to form.

Rule five states that if an approach is currently receiving a green signal and there is a stationary vehicle within a distance e from the intersection on an adjoining exit lane, then the traffic signal must be switched to terminate service. This ensures that service is terminated to an approach before the vehicles travelling along it can enter and block the intersection due to spill-back from the adjoining exit lanes.

The final rule states that if stationary vehicles are present within a distance e from the intersection along all adjoining exit lanes, then all the traffic signals should be switched to red. Like the previous rule, this rule is intended to prevent vehicles from moving into the intersection and blocking it.

A pseudo-code description of these six rules is presented in Algorithm 5.1. For each time step Algorithm 5.1 increments the amount of green time received since the last change of signal at intersection I by one time unit (line 2) and increments the counter $\varphi_I(t)$ of intersection I by the number of vehicles waiting behind or approaching a red signal within d metres of intersection I (line 3). Rule 6 ensures that all signals displayed are red if all adjoining exit lanes of intersection I are blocked by stationary vehicles (lines 4 and 5). Rule 5 changes a signal from green to red in the event that there is a stationary vehicle within a distance of e metres of intersection I along an adjoining exit lane of a lane currently receiving a green signal (line 4). Rule 4 results in a signal change if there are no vehicles approaching a green signal within d metres of intersection I and at least one vehicle waiting behind or approaching a red light within d metres of intersection I (line 16). Rule 3 prevents the tail of a platoon from being separated while allowing larger platoons to be divided. If there are fewer than s vehicles approaching a green signal within r metres of intersection I, the green signal is not changed to red until they have crossed the

Algorithm 5.1: Gershenson's self-organising traffic signal control algorithm

Input : The positions of all vehicles approaching intersection I at each time step.
Output: A signal phase switching decision for each time step.

```
for each time step t do
    t_{I_green}(t) ← t_{I_green}(t) + 1;
    φ_I(t) ← φ_I(t) + |vehicles approaching a red signal within d metres of intersection I|;
    if |vehicles stopped beyond a green signal within e metres of intersection I on
    adjoining lane| > 0 then
        if |vehicles stopped beyond a red signal within e metres of intersection I on
        adjoining lane| > 0 then
            SwitchAllSignalsToRed;
            t_{I_green}(t) ← 0;
        end
        else
            SwitchSignals;
            t_{I_green}(t) ← 0;
            φ_I(t) ← 0;
        end
    end
    else if |vehicles stopped beyond a red signal within e metres of intersection I on
    adjoining lane| = 0 then
        if φ_I(t) > 0 and |vehicles approaching a green signal within d metres of
        intersection I| = 0 then
            SwitchSignals;
            t_{I_green}(t) ← 0;
            φ_I(t) ← 0;
        end
        else if φ_I(t) > ς_I then
            if |vehicles approaching a green signal within r metres of intersection I| > s
            then
                if t_{I_green}(t) > u_I then
                    SwitchSignals;
                    t_{I_green}(t) ← 0;
                    φ_I(t) ← 0;
                end
            end
        end
        else
            KeepCurrentSignalGreen;
        end
    end
end
```

intersection (line 22). Rule 2 ensures that a signal remains green for at least u_I time steps to prevent signals switching too frequently (line 23). Provided that rules 2 and 3 are satisfied, rule 1 will change a green signal to red as soon as the counter $\varphi_I(t)$ reaches the threshold value ς_I (line 21).

5.3.2 Lämmer and Helbing's traffic signal control algorithm

The self-organising traffic control algorithm proposed by Lämmer and Helbing [54, 53] differs from that proposed by Gershenson and Rosenblueth [38] in that it comprises both an *optimisation strategy* and a *stabilisation strategy*. Traffic flow through the road network is described by a fluid dynamic model which considers vehicle flow rates rather than the characteristics of individual vehicles. The traffic dynamics of an intersection approach are characterised by the arrival rate and departure rate of vehicles to and from the intersection. Again it is assumed that all vehicles travel at a fixed, uniform speed. Intersection approach j is associated with a maximum vehicle flow rate $Q_j^{\max}$, as well as a vehicle arrival rate $Q_j^{\text{arr}}(t) \leq Q_j^{\max}$ and a vehicle departure rate[4] $Q_j^{\text{dep}}(t) \leq Q_j^{\max}$ at time t. From these characteristic vehicle flow rates, the cumulative number of vehicles *expected* to reach the intersection by time t is $N_j^{\text{exp}}(t) = \int_{-\infty}^{t} Q_j^{\text{arr}}(t' - L_j/V_j)\,\mathrm{d}t'$ (where L_j/V_j corresponds to the time required for a vehicle to traverse the length L_j of approach j at the speed limit V_j associated with approach j), while the number of vehicles which have departed from the approach by time t is $N_j^{\text{dep}}(t) = \int_{-\infty}^{t} Q_j^{\text{dep}}(t')\,\mathrm{d}t'$. The queue length (in terms of the number of vehicles) $q_j(t)$ along approach j at time t is the difference between these two cumulative values, *i.e.* $q_j(t) = N_j^{\text{exp}}(t) - N_j^{\text{dep}}(t)$. While $q_j(t)$ is capable of capturing the associated in- and out-flows of approach j, as well as the time required to resolve a vehicle queue, it does not explicitly account for the spatial location of congestion along approach j [53], rather assuming a vertical queue formation.

The algorithm takes into account the *setup time* τ_j required before a green signal may be displayed to approach j. This setup time (usually 3 to 8 seconds) typically comprises an amber and all-red period, and is determined according to safety requirements [53]. The remaining setup time that will be experienced by vehicles along approach j at time t is denoted by $\tau_j(t)$, where $0 \leq \tau_j(t) \leq \tau_j$. Traffic models which do not explicitly incorporate vehicle accelerations and decelerations adjust this setup time in an attempt to implicitly account for delays caused by driver reactions and finite acceleration times, as is the case with the model of Lämmer and Helbing [53]. This practice has been shown to lead to discrepancies in the accuracy of the actual delay times experienced by vehicles [13].

The optimisation strategy relies on the effective anticipation of traffic flows and platoons in order to predict the effects on future delay times of vehicles in terms of starting, continuing, or terminating service processes [52]. This anticipation is employed to determine the amount of green time $\hat{g}_j(t)$ required to clear all $q_j(t)$ currently queued vehicles as well as those which join the queue during the remaining set-up phase and while the queue is being cleared. The queue will therefore be fully dissipated at time $t + \tau_j(t) + \hat{g}_j(t)$ when the total number of vehicles which have departed from the intersection equals the total number of vehicles that have arrived at the intersection, *i.e.*

$$N_j^{\text{dep}}(t) + \hat{g}_j(t) Q_j^{\max} = N_j^{\text{exp}}(t + \tau_j(t) + \hat{g}_j(t)). \tag{5.8}$$

Here, the value of $\hat{g}_j(t)$ is taken to be the largest possible solution of (5.8). The second value on the left hand side of (5.8) is the number of vehicles expected to cross the intersection at the maximum flow rate and is denoted by $\hat{n}_j(t) = \hat{g}_j(t) Q_j^{\max}$. This value is assumed to incorporate

[4] It is assumed that queued vehicles depart from an intersection at the maximum flow rate.

all currently queued vehicles, all vehicles arriving during the setup phase and while the queue is being cleared, as well we as all vehicles arriving as a platoon immediately after the queue is cleared. The optimisation strategy is to assign approach j a *priority index*

$$o_j(t) = \frac{\hat{n}_j(t)}{\tau_\sigma + \tau_j(t) + \hat{g}_j(t)}. \tag{5.9}$$

In (5.9), τ_σ is the *penalty* incurred for terminating service to approach σ currently receiving service and reflects the setup time necessary before service is resumed to approach σ. In the case that $j = \sigma$, this penalty term falls away. This priority index may be interpreted as the average service rate of the approach, *i.e.* the anticipated number of vehicles $\hat{n}_j(t) = \hat{g}_j(t)Q_j^{\max}$ expected to be served during a time period of length $\tau_j(t) + \hat{g}_j(t)$ [53]. The approach achieving the largest priority index value is awarded service. A pseudo-code description of the optimisation strategy is provided in Algorithm 5.2.

Algorithm 5.2: Lämmer and Helbing's optimisation strategy

Input : The positions and speeds of all vehicles approaching the intersection at each time step.

Output: A signal phase switching decision for each time step.

```
1  for each time step t do
2      for each traffic flow j do
3          if σ(t) = j then
4              ĝ_j(t) ← ĝ_j(t) − 1;
5              o_j(t) = n̂_j(t) / (τ_j(t) + ĝ_j(t));
6          end
7          else if σ(t) ≠ j then
8              ĝ_j(t) ← q_j(t)/Q_j^max + q_j(t)Q_j^arr(t) / (Q_j^max)^2 + τ_j Q_j^arr(t) / Q_j^max;
9              o_j(t) = n̂_j(t) / (τ_σ + τ_j(t) + ĝ_j(t));
10             if o_j(t) > o_σ(t) then
11                 σ(t) ← j
12             end
13             else
14                 σ(t) ← σ(t − 1)
15             end
16         end
17     end
18 end
```

At each time step, Algorithm 5.2 determines whether or not approach j is currently receiving service (line 3). If this is true (*i.e.* $\sigma(t) = j$) then the remaining green time is decremented (line 4) and the priority index of approach j is calculated without the penalty term τ_σ (line 5). If approach j is not currently receiving service, then the green time required to clear all anticipated arriving vehicles is calculated (line 8) as well as the priority index with the additional penalty term τ_σ (line 9). The algorithm then compares the priority indices of all the approaches (line 10) and if the priority index value of an approach not currently receiving service exceeds that of one which is, a signal change is made (line 11) otherwise the approach currently receiving service continues to do so (line 14).

Lämmer and Helbing [53] couple the aforementioned optimisation strategy with a *stabilisation*

strategy so as to ensure stability[5] along all intersection approach lanes. This stabilisation strategy is implemented should the local optimisation strategy become *unstable*, in which case it is deemed to be potentially inferior to a fixed-time control scheme which would ensure stability under the same prevailing traffic conditions [54]. The stabilisation strategy ensures stability by assigning service to approach j as soon as a critical number of vehicles n_j^{crit} can be served at the maximum flow rate $Q_j^{\max}$. This critical number of vehicles is determined according to a pre-defined *threshold function*.

The choice of the threshold function has a major impact on the regularity of service intervals in terms of frequency and duration. The derivation of this threshold function requires the average arrival rate of vehicles along all intersection approaches as well as the cycle length and green times of a stable fixed-time control programme associated with these average arrival rates. The algorithm of Lämmer and Helbing [53] assumes the values of all of these parameters to be known and available. The stabilisation strategy attempts to ensure that at least as many vehicles are served by the intersection as would be the case, on average, if a stable, fixed-time control programme were to be implemented. The stabilisation strategy functions according to two key requirements. The first requirement is that each traffic flow should be served once *on average* within a desired service interval of length Z. Here, Z is the cycle time of an associated stable, fixed-time control programme. This requirement ensures a desired regularity of service to each traffic flow. The second requirement is that each traffic flow should be served *at least* once within a service interval of length $Z^{\max}$. Here, $Z^{\max} \geq Z$ [6] is a maximum allowable red time. This ensures that no traffic flow is required to wait for service for longer than a period of length $Z^{\max}$.

The algorithm anticipates the service interval $z_j(t)$ for approach j which corresponds to the time interval between the end of the last green time received by approach j and the start of the next service period it will receive. The critical threshold of vehicles for approach j may then be determined as a function

$$n_j^{\text{crit}}(z_j(t)) = Q_j^{\text{arr}}(t) Z \frac{Z^{\max} - z_j(t)}{Z^{\max} - Z} \tag{5.10}$$

of this service interval. The function in (5.10) meets the first requirement since the number of vehicles expected to arrive within the desired service interval of length Z is $Q_j^{\text{arr}}(t)Z$. This is equal to the critical threshold value for the case in which $z_j(t) = Z$ (approach j receives service as soon as there are as many vehicles to serve as would arrive on average during the desired service interval of length Z). The second requirement is also satisfied since $n_j^{\text{crit}}(z_j(t)) \leq 0$ if $z_j(t) \geq Z^{\max}$ (as soon as as the anticipated service interval of an approach is expected to exceed the maximum allowable red time, the approach is scheduled to receive service). As soon as an approach is identified as being critical (*i.e.* when $\hat{n}_j(t) \geq n_j^{\text{crit}}(z_j(t))$), the phase during which the approach receives service is moved forward as soon as possible and is added to the back of an ordered set Υ containing critical approaches requiring immediate service. An approach remains in a critical state until either the vehicle queue along it has been cleared completely or until it has received a green time equal to that which a stable, fixed-time control programme would have awarded it (*i.e.* $Z\, Q_j^{\text{arr}}(t)/Q_j^{\max}$), at which point it is removed from Υ. If it so happens that more than one flow is considered to be critical at the same time, then the flows are served in the order in which they became critical. A pseudo-code description of this stabilisation strategy is presented in Algorithm 5.3.

At each time step, Algorithm 5.3 determines whether the expected number of vehicles requiring

[5] A controlled queueing network is deemed to be stable if all queue lengths remain bounded at all times [81].

[6] The greater the magnitude of difference between $Z^{\max}$ and Z, the more flexible the control algorithm is. If, on the other hand, $Z^{\max} = Z$, the functionality of the algorithm will be the same as that of a fixed-time control strategy with a cycle time of Z.

Algorithm 5.3: Lämmer and Helbing's stabilisation strategy

Input : The positions and speeds of all vehicles approaching the intersection at each time step.

Output: A signal phase switching decision for each time step.

```
1  for each time step t do
2      for each traffic flow j do
3          if n̂_j(t) ≥ n_j^crit(z_j(t)) then
4              ϒ ← ϒ ∪ {j};
5          end
6          else
7              ϒ ← ϒ
8          end
9          if j is the first element of ϒ then
10             if q_j(t) = 0 or g_j(t) = Z Q_j^arr(t) / Q_j^max then
11                 ϒ ← ϒ\{j};
12             end
13             else
14                 ϒ ← ϒ
15             end
16         end
17     end
18     σ(t) = first element of ϒ;
19 end
```

service exceeds some predetermined threshold value (line 3). If this is the case, then approach j is added to the back of Υ to receive service as soon as possible (line 4). Once the queue of vehicles along approach j has been dissipated or it has received a green signal for a time period of at least $Z\,Q_j^{\text{arr}}(t)/Q_j^{\max}$ seconds (line 10), approach j is removed from Υ (line 11). At each time step, service is awarded to the first element of Υ (line 18).

The optimisation and stabilisation strategies described above are combined as follows: The optimisation strategy is responsible for assigning service to the various phases as long as none of the approaches served during any of the phases is considered to be critical. A pseudo-code description of Lämmer and Helbing's final self-organising traffic signal control algorithm is presented in Algorithm 5.4.

At each time step, Algorithm 5.4 determines whether there are any critical approaches requiring immediate service (line 2). If there are, the approach which has been critical the longest (*i.e.* the first element of Υ) receives service (line 3). If there are no critical approaches requiring immediate service, then service is assigned to the approach associated with the maximum priority index (line 5).

5.3.3 Algorithmic appraisal

The two self-organising traffic control strategies described in §5.3.1 and §5.3.2 have been shown to outperform both optimised classical fixed-time control programs [27, 37, 38, 53, 54, 103] and state-of-the-art centralised traffic responsive systems [54] in terms of minimising vehicle delay. They are, however, not without limitations. Both algorithms rely on the appropriate selection of

Algorithm 5.4: Lämmer and Helbing's self-organising traffic signal control algorithm

Input : The positions and speeds of all vehicles approaching the intersection at each time step.

Output: A signal phase switching decision for each time step.

```
for each time step t do
    if ϒ ≠ ∅ then
        σ(t) = first element of ϒ;
        else
            σ(t) = arg max_j o_j(t)
        end
    end
end
```

several parameter values in order for them to function effectively. The algorithm of Gershenson and Rosenblueth [38] contains seven parameters and while the optimisation strategy of Lämmer and Helbing [53] is free of parameters, their stabilisation strategy relies on the appropriate selection of two parameters, Z and $Z^{\max}$, which are chosen according to a stable, fixed-time control strategy based on average vehicle arrival and departure rates. While Gershenson and Rosenblueth [38] maintain that their algorithm is "robust" in that it is "not affected by modest parameter variations," they do not provide a procedure for selecting appropriate parameter values. The author considers this a vital drawback as it has been shown that appropriate parameter selection is integral to the effectiveness of self-organising traffic control algorithms [28]. Lämmer and Helbing [53], on the other hand, provide a method for parameter selection for their stabilisation strategy, but their method relies heavily on an assumed stable, fixed-time control programme designed for constant average traffic flows which they do not elaborate upon. A potential drawback of this approach is that the stabilisation strategy may therefore be subject to the same limitations as the fixed-time control approach on which it is based in that it may not be flexible enough to adapt adequately to sudden changes in traffic demand.

5.4 Chapter summary

A brief overview of existing traffic signal control techniques was presented in §5.1. An example of a fixed-time traffic signal control methodology was described in §5.2.1 for equalising the saturation along critical intersection approaches. A second fixed-time traffic signal control methodology for facilitating the movement of green-waves through a traffic network was described in §5.2.2. Two examples of self-organising traffic signal control algorithms were finally presented in §5.3. A discussion was also included on the advantages and disadvantages of these self-organising algorithms.

CHAPTER 6

Three novel self-organising traffic signal control algorithms

Contents

This chapter serves to introduce and describe three novel, decentralised, self-organising traffic signal control algorithms. The first of the three algorithms is presented in §6.1 and is inspired by inventory control techniques. The second novel traffic signal control algorithm introduced is described in §6.2 and is inspired by the chemical process of osmosis. The third and final traffic signal control algorithm introduced in this chapter is described in §6.3. This is a hybrid algorithm which combines the previous two algorithms and supplements them with an intersection utilisation maximisation technique.

6.1 A traffic control algorithm inspired by inventory theory

The well-known *economic order quantity* (EOQ) in inventory theory embodies two key decision variables: determining inventory reorder points in time and determining the associated reorder quantities that minimise the total costs incurred in maintaining inventory levels and failing to meet customer demand. Parallels may be drawn between this inventory control problem and the problem of allocating green time to the various phases of a traffic signal cycle. Green time may be considered as the product in inventory, while the phases of the signal cycle and their associated approach lanes may be interpreted as customers seeking to replenish their green-time stock. The vehicles served during the specific phases may be viewed as the cause of demand for inventory stock. In this analogy, the two key inventory theory variables mentioned above translate to the following questions: When should a particular phase receive green time? How much green time should this phase receive?

6.1.1 The costs involved in basic inventory control models

In inventory theory, all costs are typically expressed in monetary terms, while in the context of traffic control at signalised intersections, it is more natural to express costs in terms of vehicle delay time. The following four costs are usually associated with basic inventory models [100]: the *ordering and setup cost*, the *unit purchasing cost*, the *holding* or *carrying cost*, and the *stockout* or *shortage cost.*

The ordering and setup cost is incurred when placing an order for inventory stock, or setting up a production run if the stock is produced internally. Costs of this nature usually do not depend on the size of the order or length of the production run; they are typically fixed costs incurred each time stock is ordered or a production run is initiated. In terms of traffic signal control, the ordering and setup cost may be interpreted as the unavoidable delay time incurred by vehicles waiting at an intersection during the necessary amber and all-red phases preceding the onset of a green signal.

The unit purchasing cost is the variable cost associated with purchasing or producing a single unit of stock and typically includes all labour costs, overhead costs, raw material costs and shipping costs. There is no equivalent interpretation for unit purchasing cost in the context of traffic signal control.

The holding or carrying cost measures the expenses associated with holding one unit of stock in inventory for a single time period. This cost normally incorporates storage cost, insurance cost and inventory taxes. The most significant component of holding cost, however, is the opportunity cost associated with tying up capital in inventory. In the context of traffic signal control, a certain phase may be considered to be holding inventory while it is receiving service. This cost may be interpreted as the delay time incurred by all vehicles at an intersection which are required to wait while the particular phase receives service (*i.e.* all vehicles along intersection approaches not currently receiving service).

The stockout or shortage cost is the result of demand not being met on time and is comparatively harder to quantify in inventory theory. It includes the cost of lost sales as well as the cost of loss of customer goodwill. In the context of traffic signal control, stockout cost may be interpreted as the vehicle delay time resulting from the termination of a green signal (*i.e.* the delay time experienced by all vehicles along an intersection approach to which a green signal has just been terminated while they wait for service to resume).

An example of the trade-off between holding cost and ordering cost in a basic EOQ model is shown in Figure 6.1.

6.1.2 The inventory traffic signal control algorithm

The newly proposed *inventory traffic signal control algorithm* (I-TSCA) is a self-organising, adaptive traffic signal control algorithm which seeks to minimise the total delay time experienced by vehicles passing through an intersection. It is inspired by the inventory theory analogy described above and seeks to minimise the total cost associated with basic inventory models.

The I-TSCA functions by calculating this total cost (*i.e.* the sum of the setup cost, the holding cost and the stockout cost) associated with awarding green time to each phase of the traffic signal cycle. Green time is then awarded to the phase which will result in the lowest total cost, thus minimising the total delay time of all vehicles which are to pass through the intersection. Appropriate radar detection equipment, as described in §1.2, is assumed to be mounted at the intersection, allowing for the detection and tracking of individual vehicles in terms of their

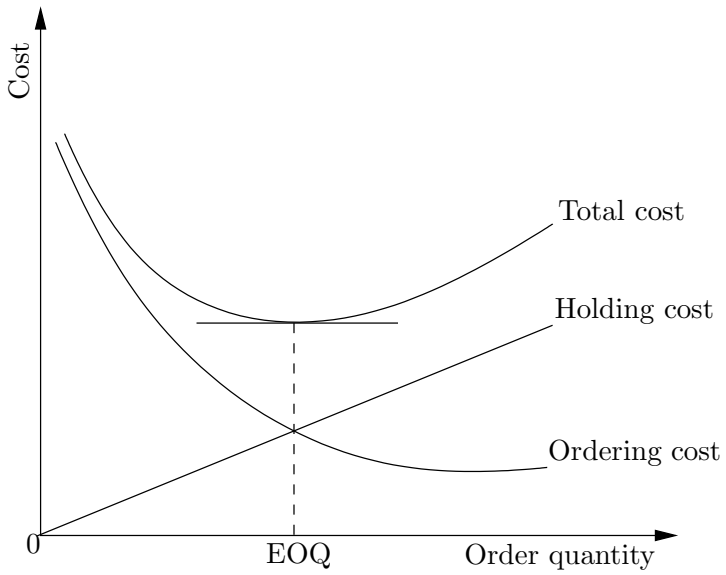


FIGURE 6.1: *An example of the trade-off between holding cost and ordering cost in the basic EOQ model.*

speeds, distances from the intersection and the headways between vehicles and the vehicles in front of them. These data are taken as input to the I-TSCA, allowing it to calculate the effective "customer demand" for each approach to the intersection and thereby for each phase of the traffic control cycle.

Three sets are associated with each lane of approach to an intersection. The first of these sets, $\mathfrak{C}_j(t)$, contains all vehicles present along approach lane j at time t. The second set, $\mathfrak{S}_j(t)$, contains all stationary, queued vehicles along approach lane j at time t and is a subset of $\mathfrak{C}_j(t)$. The third set, $\mathfrak{Q}_j(t)$, contains both currently queued vehicles and vehicles which have not yet stopped, but will become queued, either behind an existing queue, or behind an amber or red traffic signal along approach lane j at time t. To summarise, $\mathfrak{C}_j(t) \supset \mathfrak{Q}_j(t) \supset \mathfrak{S}_j(t)$. Central to determining whether or not a vehicle will become queued, is the ability to predict the location of the *queue position* $\rho_j(t)$ along approach lane j at time t. This queue position indicates how far upstream from the intersection a vehicle queue reaches along approach lane j at time t. Suppose that at time t the speed of vehicle $i \in \mathfrak{C}_j(t) \backslash \mathfrak{Q}_j(t)$ is $v_i(t)$ and that its distance to $\rho_j(t)$ is $d_{i,\rho_j(t)}(t)$, as illustrated in Figure 6.2. Should it be determined at time t that vehicle i will become queued, it is assigned a stopping point $\mu_i = \rho_j(t)$, as well as a position $\epsilon_i(t)$ in the predicted vehicle queue. The I-TSCA continually updates the predicted vehicle queue length $|\mathfrak{Q}_j(t)|$ along approach lane j. Vehicles are assumed to depart from a queue along approach lane j at a constant rate of η_j vehicles per second, where η_j is equal to the maximum flow rate of lane j, $Q_j^{\max}$ vehicles per second.

Each traffic signal cycle is assumed to comprise a set of $\mathfrak{P}$ phases and associated setup periods which precede each phase. A remaining green time $\chi_m(t)$ and a remaining setup time $\tau_m(t)$ are associated with phase m of the traffic signal control cycle. Suppose that vehicle i is travelling along intersection approach lane j which receives service during phase m of the traffic signal cycle. To determine whether or not vehicle i will become queued, and hence added to $\mathfrak{Q}_j(t)$, the I-TSCA assesses whether the time it will take vehicle i to reach $\rho_j(t)$ is either less than the sum of the remaining red time together with the time required to clear the current predicted queue (if the traffic signal displayed is not green) or less than the time required to clear the current predicted queue of vehicles (if the signal displayed is green). If this is indeed the case, then vehicle i will become queued, hence delayed and thus added to $\mathfrak{Q}_j(t)$. In other

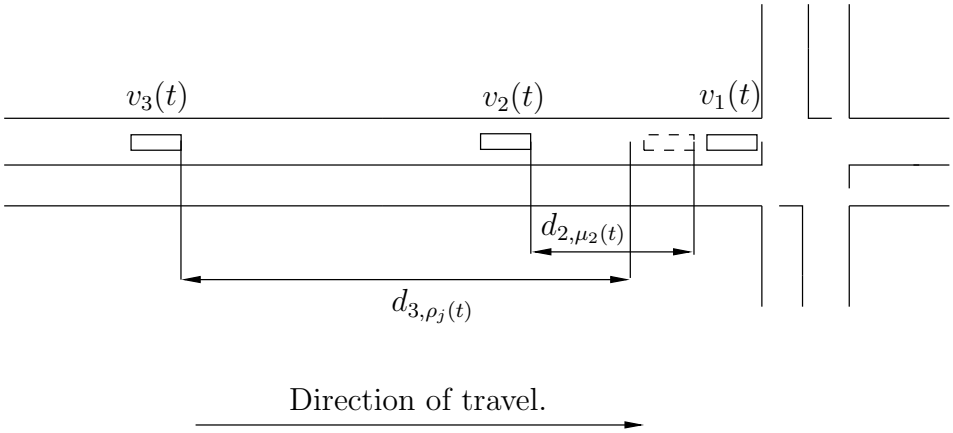


FIGURE 6.2: *Individual vehicle characteristics. At time t, vehicle 1 is waiting at the intersection* (i.e. $v_1(t) = 0$) *along an approach lane. Vehicle 2 will become queued behind vehicle 1 within $d_{2,\mu_2(t)}/v_2$ seconds. Vehicle 3 is $d_{3,\rho_j(t)}$ metres from the currently predicted queue position.*

words, if the traffic signal currently displayed at time t is not green and $d_{i,\rho_j(t)}(t)/v_i(t) < \sum_{p\in\mathfrak{P}\setminus\{m\}}(\chi_p(t)+\tau_p(t)) + |\mathfrak{Q}_j(t)|/\eta_j$, then vehicle i is immediately added to $\mathfrak{Q}_j(t)$ and $\rho_j(t)$ is incremented by the *effective length* $\hat{\ell}_i$ of vehicle i (*i.e.* the actual length ℓ_i of vehicle i together with the minimum space gap required between stationary vehicles). Similarly, if the traffic signal displayed at time t is green and $d_{i,\rho_j(t)}(t)/v_i(t) < |\mathfrak{Q}_j(t)|/\eta_j$, then again vehicle i is immediately added to $\mathfrak{Q}_j(t)$ and $\rho_j(t)$ is incremented by $\hat{\ell}_i$. In the case where $\mathfrak{Q}_j(t)$ is empty (*i.e.* $\rho_j(t)$ is located at the stop line of the intersection) vehicle i is added to $\mathfrak{Q}_j(t)$ under one of two conditions: If the traffic signal displayed at time t is green and $d_{i,\rho_j(t)}(t)/v_i(t) > \chi_m(t)$ or if the traffic signal displayed at time t is not green and $d_{i,\rho_j(t)}/v_i(t) < \sum_{p\in\mathfrak{P}\setminus\{m\}}(\chi_p(t)+\tau_p(t))$.

The I-TSCA iterates through three steps. The first step is the calculation of the required green time for each phase of the traffic control cycle. During the second step, the associated costs (in terms of vehicle delay time) are calculated for each phase of the traffic control cycle, should the previously calculated green time be implemented. The total cost of all the possible phase options are compared during the third step, and the phase achieving the lowest total cost is selected for service. More detailed descriptions follow of how these steps are performed.

Step 1: Calculating the required green time of a phase. The I-TSCA calculates the required green time $\gamma_j(t)$ for approach lane j as the amount of time required to clear all vehicles contained in $\mathfrak{Q}_j(t)$. If it so happens that $\mathfrak{Q}_j(t)$ is empty, then the required green time of approach lane j is taken as the time it would take for the front vehicle along the lane to clear the intersection. The required green time for approach lane j is therefore

$$\gamma_j(t) = \begin{cases} d_{1,\rho_j(t)}(t)/v_1(t) & \text{if } |\mathfrak{Q}_j(t)| = 0 \text{ and } |\mathfrak{C}_j(t)| > 0, \\ |\mathfrak{Q}_j(t)|/\eta_j & \text{if } 0 < |\mathfrak{Q}_j(t)| < \infty \text{ and } |\mathfrak{C}_j(t)| > 0, \\ \infty & \text{if } |\mathfrak{C}_j(t)| = 0. \end{cases} \tag{6.1}$$

For the case where $|\mathfrak{Q}_j(t)| = 0$ in (6.1), $d_{1,\rho_j(t)}(t)$ is simply the distance between the front vehicle along approach lane j and the intersection. The reason for selecting the vehicle closest to the intersection when determining the required green time is to maximise intersection utilisation. If $\mathfrak{A}_m$ is the set of all approach lanes served during phase m of the traffic signal cycle, then the required green times for lane j are compared, and the minimum required green time over all the lanes is selected as the required green time for phase m, *i.e.* $\Gamma_m(t) = \min_{j\in\mathfrak{A}_m}\gamma_j(t)$. The reason for selecting the minimum green time as representative for the phase is to ensure that

a lane associated with a relatively large green time does not prevent other lanes with shorter associated green times from receiving service, since providing a large amount of green time may prove to be uneconomical in terms of minimising the total delay of all vehicles utilising the intersection. For this reason, if there are no vehicles approaching the intersection along a lane, then the associated green time of that lane is set to infinity so as to ensure that the lane is not considered for receiving a green signal.

Step 2: Calculating the associated vehicle delay. If it is determined that vehicle i along approach lane j will become queued, the I-TSCA calculates an expected delay term $\phi_{ij}(t)$, indicating how long this vehicle may expect to be delayed. This delay term depends on whether or not approach lane j is receiving service, indicated by a binary parameter $\kappa_j(t)$, where $\kappa_j(t) = 1$ if approach lane j is receiving service at time t, or $\kappa_j(t) = 0$ otherwise. If approach lane j is served during phase m of the traffic control cycle, then

$$\phi_{ij}(t) = \begin{cases} \tau_m(t) + \frac{\epsilon_i(t)}{\eta_j} - \frac{d_{i,\mu_i}(t)}{v_i(t)} & \text{if } \kappa_j(t) = 1, \\ \sum_{p\in\mathfrak{P}} \tau_p(t) + \sum_{p\in\mathfrak{P}\setminus\{m\}} \chi_p(t) - \frac{d_{i,\mu_i}(t)}{v_i(t)} + \frac{\epsilon_i(t)}{\eta_j} & \text{if } \kappa_j(t) = 0. \end{cases} \tag{6.2}$$

For the case where $\kappa_j(t) = 1$ in (6.2), the first term in the expression for $\phi_{ij}(t)$ is the remaining setup time at time t before a green signal is displayed to approach lane j during phase m. The second term is the amount of time it will take for the queue of vehicles in front of vehicle i (at position $\epsilon_i(t)$ in the queue) to dissipate at a rate of η_j vehicles per second. The third term is the time that will elapse before the vehicle comes to rest in the queue. If the vehicle is already stationary, $d_{i,\mu_i}(t) = 0$ and this last term falls away. For the case where $\kappa_j(t) = 0$, the first three terms represent the time that vehicle i will be required to wait before a green signal is displayed to approach lane j and equals the sum of all remaining setup times of the traffic control cycle, as well as the remaining green times of all the other phases of the traffic control cycle, less the time it will take for vehicle i to become queued. The last term is again the amount of time vehicle i will be delayed while the queue in front of it dissipates upon receiving a green signal. In the case where vehicle i is not predicted to become queued, it will not be delayed and will therefore have an associated delay term of $\phi_{ij}(t) = 0$. The set $\mathfrak{J}$ contains all approach lanes to the intersection. To calculate the cost $\Phi_m(t)$ associated with implementing the required green time for phase m, the I-TSCA sums the delay terms of all detected vehicles approaching the intersection, *i.e.* $\Phi_m(t) = \sum_{j\in\mathfrak{J}} \sum_{i\in\mathfrak{C}_j(t)} \phi_{ij}(t)$. The setup cost is accounted for by summing the required setup time $\tau_m(t)$ for each vehicle that will be delayed during this period. The holding cost is accounted for by summing the delay terms for each vehicle along all lanes j for which $\kappa_j(t) = 0$. The stockout cost is accounted for by summing the delay terms of vehicles along all lanes j currently receiving service, should $\kappa_j(t) = 0$ (*i.e.* service be terminated) at time t.

Step 3: Assigning service. Service is assigned to the phase m for which $\Phi_m(t)$ is a minimum. If phase m is selected for service at time t, it is assigned a green time of $\Gamma_m(t)$. After assigning service to phase m, the I-TSCA continues to re-evaluate required green times and associated delay costs, but service is not switched until the assigned green time of $\Gamma_m(t)$ time units has elapsed. This is to prevent too frequent switching between phases. Once the assigned green time has elapsed (*i.e.* $\chi_m(t) = 0$), the I-TSCA once again compares all $\Phi_m(t)$ values for $m \in \mathfrak{P}$, assigning service to the phase achieving the minimum resulting delay cost. It may occur that the phase which has just received service continues to be associated with the minimum resulting delay cost in which case it is assigned a new, extended green time and continues to receive service.

A pseudo-code description of the I-TSCA is presented in Algorithm 6.1. At each time step the algorithm calculates the required green time of each intersection approach lane as described

in Step 1 above (lines 3 to 10). The algorithm then calculates the expected delay of each vehicle approaching the intersection as described in Step 2 above (lines 12 to 19). After having calculated the green times of intersection approach lanes and the expected delay of all vehicles travelling along these lanes, the algorithm assigns the required green time of each phase of the control cycle (line 22) and the expected delay time associated with this required green time should it be implemented (line 23). Service is then awarded to the phase which will result in the lowest overall delay (lines 24 to 26), provided that the remaining green time of the phase which was previously receiving service is zero (line 24).

An example of the I-TSCA implemented at an intersection within the microscopic traffic simulation modelling framework described in §4.1 is depicted in Figure 6.3. The line labelled $\Phi_1(t)$ corresponds to the delay time associated with awarding a green signal to vehicles travelling in a north-to-south or south-to-north direction at time t, while the line labelled $\Phi_2(t)$ corresponds to the delay time associated with awarding a green signal to vehicles travelling in a west-to-east or east-to-west direction at time t.

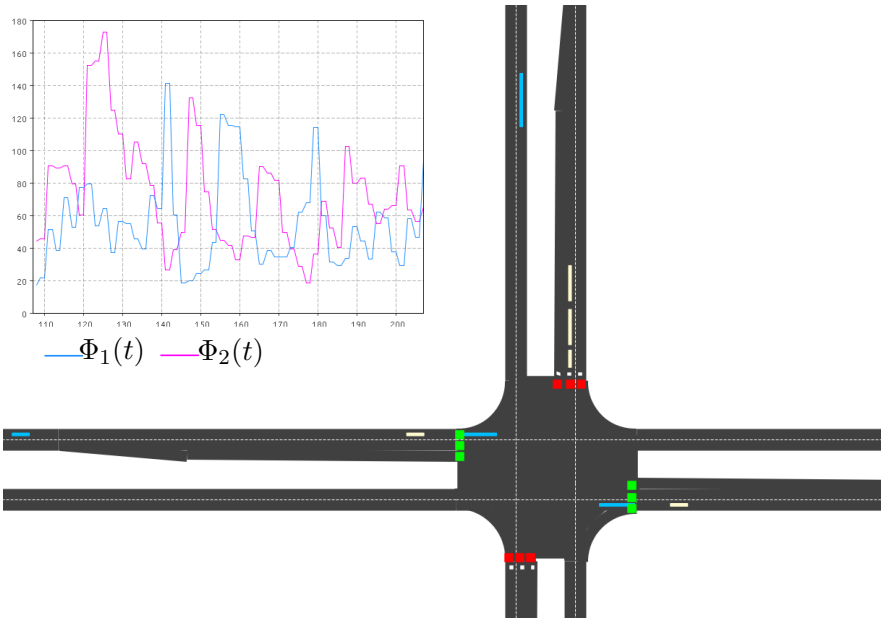


FIGURE 6.3: *An example of the I-TSCA implemented at an intersection within the microscopic traffic simulation modelling framework described in §4.1.*

6.2 A traffic control algorithm inspired by osmosis

When two liquids of differing solute concentrations are separated by a partially permeable membrane through which the solute cannot pass, the solvent passes by osmosis from the liquid with the lower solute concentration, through the membrane, and into the liquid with a higher solute concentration, as illustrated in Figure 6.4 [11, 76]. This movement is said to occur down a *concentration gradient* [89]. The liquid with a high water concentration and thus lower solute concentration has a higher *osmotic potential* or *osmotic pressure* [11] than that of the liquid with a higher solute concentration and solvent molecules continue to move from a region of high osmotic pressure to a region of low osmotic pressure until the osmotic pressures of the liquids on both sides of the membrane are equal [89].

The notion of osmotic pressure has inspired the second novel traffic control algorithm proposed

Algorithm 6.1: The inventory traffic signal control algorithm

Input : The positions and speeds of all vehicles approaching the intersection at each time step.

Output: A signal phase switching decision for each time step.

for *each time step* t **do**
  for *each approach lane* j *of the intersection* **do**
    if $|\mathfrak{Q}_j(t)| = 0$ **and** $|\mathfrak{C}_j(t)| > 0$ **then**
      $\gamma_j(t) = d_{1,\rho_j(t)}(t)/v_1(t)$;
    end
    else if $0 < |\mathfrak{Q}_j(t)| < \infty$ *and* $|\mathfrak{C}_j(t)| > 0$ **then**
      $\gamma_j(t) = |\mathfrak{Q}_j(t)|/\eta_j$;
    end
    else if $|\mathfrak{C}_j(t)| = 0$ **then**
      $\gamma_j(t) = \infty$;
    end
    for *each vehicle* i *on approach lane* j **do**
      if $\kappa_j(t) = 1$ **then**
        $\phi_{ij}(t) = \tau_m(t) + \frac{\epsilon_i(t)}{\eta_j} - \frac{d_{i,\mu_i}(t)}{v_i(t)}$;
      end
      else if $\kappa_j(t) = 0$ **then**
        $\phi_{ij}(t) = \sum_{p\in\mathfrak{P}} \tau_p(t) + \sum_{p\in\mathfrak{P}\setminus\{m\}} \chi_p(t) - \frac{d_{i,\mu_i}(t)}{v_i(t)} + \frac{\epsilon_i(t)}{\eta_j}$;
      end
    end
  end
  for *each signal phase* m **do**
    $\Gamma_m(t) = \min_{j\in\mathfrak{A}_m} \gamma_j(t)$;
    $\Phi_m(t) = \sum_{j\in\mathfrak{J}} \sum_{i\in\mathfrak{C}_j(t)} \phi_{ij}(t)$;
    if $\Phi_m(t) < \Phi_{m'}(t)$ **and** $\chi_m(t) = 0$ **then**
      $\kappa_j(t) \leftarrow 1 \forall j \in \mathfrak{A}_m$;
      $\chi_m(t) \leftarrow \Gamma_m(t)$;
    end
  end
end

in this dissertation. In the context of traffic control at a signalised intersection, the solvent may be likened to the vehicles travelling along the intersection approach lanes while the solute may be considered as the space along the lanes not occupied by vehicles. The intersection itself may be considered as the partially permeable membrane. Each approach lane j to an intersection is coupled with an *exit lane* j' leading away from the intersection. The presence of vehicles requiring service along an approach lane j may be considered to exert a *push pressure* on the intersection which effectively pushes vehicles through the intersection, in much the same way as a liquid with a high osmotic pressure would force solvent through a membrane into a liquid of lower osmotic pressure. The presence of space on an exit lane j', on the other hand, would exert a *pull pressure* on the intersection which effectively pulls vehicles through the intersection, in much the same way as a liquid with a high solute concentration (and thus low osmotic pressure) would draw solvent through a membrane from a liquid of higher osmotic pressure.

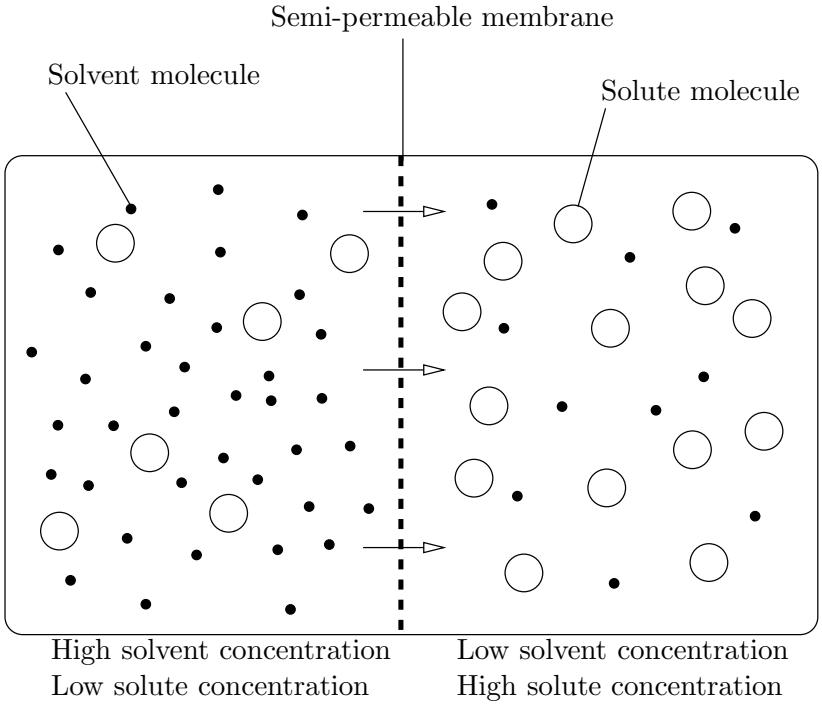


FIGURE 6.4: *The osmotic process.*

The newly proposed *osmosis traffic signal control algorithm* (O-TSCA) does not attempt to minimise vehicle delay time at a signalised intersection explicitly, nor does it factor vehicle delay time into any algorithmic calculations when considering which phase should receive service. Instead, the O-TSCA relies on the sum of push and pull pressures exerted on an intersection by respective pairs of approach and exit lanes. As with the I-TSCA, the O-TSCA assumes the installation of suitable radar detection sensors at the intersection so that vehicles approaching the intersection along lane j and departing from it along the associated exit lane j' may be detected. Physical vehicle lengths are taken as input data to the O-TSCA, as well as the space which is not occupied by vehicles along the intersection approaches.

Suppose the length of approach lane j is α_j (in metres) and that vehicle i travelling along lane j has an *effective length* $\hat{\ell}_i$, which is the actual length of the vehicle together with a minimum safety gap that has to be maintained between stationary vehicles (see Figure 6.5). At time t the O-TSCA calculates a *demand* $\delta_j(t)$ associated with lane j and an *availability* $\omega_j(t)$ associated with lane j. The demand $\delta_j(t) = \sum_{i \in \mathfrak{C}_j(t)} \hat{\ell}_i$ is a measure of the total effective space occupied by the vehicles travelling along approach lane j at time t. The availability $\omega_j(t) = \alpha_{j'} - \delta_{j'}(t)$ represents the amount of space available for vehicles currently travelling along approach lane j to occupy once they have crossed the intersection onto the adjoining exit lane j'. The *pressure* $\pi_j(t)$ exerted by approach lane j at time t on an intersection, is the sum of its demand and availability at time t, *i.e.* $\pi_j(t) = \delta_j(t) + \omega_j(t)$. The implications of calculating the pressure exerted by an approach lane in such a manner are summarised in Table 6.1. It may be the case that a vehicle passes from approach lane j through the intersection onto one of any of a number of possible exit lanes. In this scenario, the exit lane j' associated with approach lane j is taken to be the exit lane with the least space available along it. This is done so as to avoid the possibility of an approach lane receiving service in spite of insufficient space to accommodate vehicles on the adjoining exit lane j'. An adjoining exit lane j' with little space available will not contribute significantly to the pressure exerted by approach lane j on the intersection and

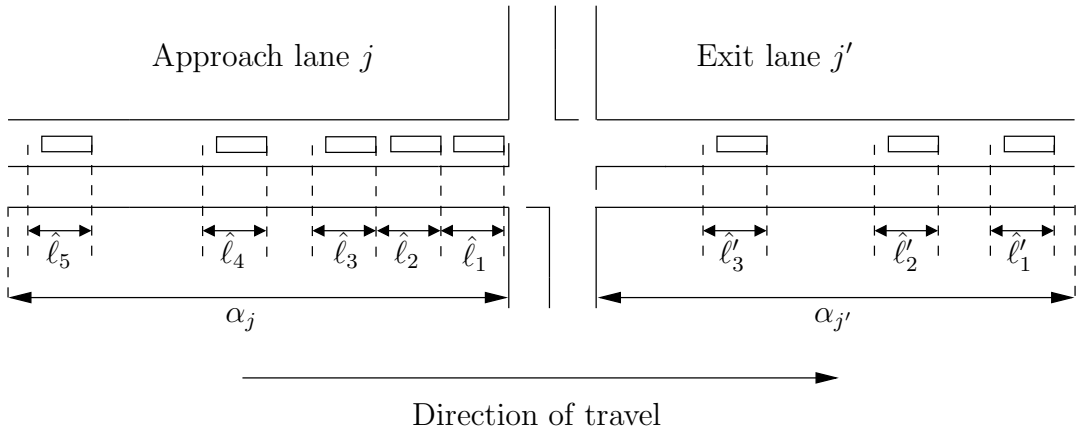


FIGURE 6.5: *Calculating the demand $\delta_j(t)$, the availability $\omega_j(t)$ and the pressure $\pi_j(t)$ of approach lane j at time t. At time t, $\delta_j(t) = \sum_{i\in\mathfrak{C}_j(t)} \hat{\ell}_i = \sum_{i=1}^{5} \hat{\ell}_i$ and $\omega_j(t) = \alpha_{j'} - \delta_{j'}(t) = \alpha_{j'} - \sum_{i\in\mathfrak{C}_{j'}(t)} \hat{\ell}'_i = \sum_{i=1}^{3} \hat{\ell}'_i$. The pressure exerted by approach lane j at time t is therefore $\pi_j(t) = \delta_j(t) + \omega_j(t) = \sum_{i=1}^{5} \hat{\ell}_i + \alpha_{j'} - \sum_{i=1}^{3} \hat{\ell}'_i$.*

so the approach lane will not carry a high priority for service. While approach lane j is receiving service, the O-TSCA calculates the *throughput* $\theta_j(t)$ of the lane. This is the total space occupied by the vehicles which have passed through the intersection from lane j onto all adjoining exit lanes.

$\delta_j(t)$	$\omega_j(t)$	$\pi_j(t)$
High	High	High
High	Low	Medium
Low	High	Medium
Low	Low	Low

Table 6.1: *Lane pressures relative to demand and availability. Both the demand and availability of approach lane j results in an effective* pushing *and* pulling *of vehicles through an intersection. A large demand results in pressure which effectively pushes vehicles through the intersection (provided that there is sufficient space available along the adjoining exit lanes to meet the demand) while a large availability effectively pulls vehicles through the intersection (provided that there is sufficient demand to fill the availability).*

To determine the pressure $\Pi_m(t)$ exerted by phase m of a traffic signal cycle at time t, the O-TSCA sums together the pressures exerted by all the lanes served during phase m, *i.e.* $\Pi_m(t) = \sum_{j\in\mathfrak{A}_m} \pi_j(t)$. The phase with the largest pressure is awarded service. If phase m was most recently selected for service at time t^*, the combined demand and availability of all the approach lanes $j \in \mathfrak{A}_m$ at time t^* are stored as the variables $\Delta_m = \sum_{j\in\mathfrak{A}_m} \delta_j(t^*)$ and $\Omega_m = \sum_{j\in\mathfrak{A}_m} \omega_j(t^*)$, respectively. Phase m continues to receive service until either

$$\sum_{j\in\mathfrak{A}_m} \theta_j(t) \geq \Delta_m \tag{6.3}$$

(*i.e.* the cumulative throughput of all the approach lanes served during phase m is at least as large as the cumulative demand of all the approach lanes served during phase m at time t^*) or

$$\sum_{j\in\mathfrak{A}_m} \theta_j(t) \geq \Omega_m \tag{6.4}$$

(*i.e.* the cumulative throughput of all approach lanes served during phase m is at least as large as the cumulative availability associated with lanes served during phase m at time t^*). Condition (6.3) ensures that all vehicles which were initially detected as requiring service do, in fact, receive service, while condition (6.4) ensures that no phase provides service to vehicles which cannot be accommodated on adjoining exit lanes due to a lack of availability of space. When either condition (6.3) or (6.4) holds, the O-TSCA again compares the pressures exerted by each phase, assigning service to the phase exhibiting the largest pressure. It may well occur that the phase which has just received service continues to achieve the largest pressure, in which case the phase will continue to receive service until the pressure of another phase exceeds it. If service is terminated to phase m at time t, the throughputs of all approach lanes j served during phase m are reset to zero (*i.e.* $\theta_j(t) = 0$ for all $j \in \mathfrak{A}_m$).

A pseudo-code description of O-TSCA is presented in Algorithm 6.2. At each time step, the algorithm calculates the demand of each approach lane to the intersection (line 3) and the availability of each approach lane (line 4). From these two values the algorithm calculates the pressure exerted on the intersection by each approach lane (line 5). For each phase of the traffic signal cycle, the algorithm calculates the total pressure exerted on the intersection by all approach lanes served during the phase (line 8). The algorithm then determines whether the throughput of the current phase receiving service is equal to either the initial demand or the initial availability of the approaches served during the current phase (line 9). If so, then the algorithm determines which phase exerts the largest pressure on the intersection (line 10). At that point in time the algorithm sets the combined demand and availability of all approach lanes that are due to receive service (lines 12 and 13, respectively). The algorithm then resets the throughputs of all approach lanes back to zero before service commences (line 14).

Algorithm 6.2: The osmosis traffic signal control algorithm.

Input : The physical lengths of all vehicles approaching the intersection at each time step.
Output: A signal phase switching decision for each time step.

1 **for** *each time step* t **do**
2 **for** *each approach lane* j *of the intersection* **do**
3 $\delta_j(t) \leftarrow \sum_{i \in \mathfrak{C}_j(t)} \hat{\ell}_i$;
4 $\omega_j(t) \leftarrow \alpha_{j'} - \delta_{j'}(t)$;
5 $\pi_j(t) \leftarrow \delta_j(t) + \omega_j(t)$;
6 **end**
7 **for** *each signal phase* m **do**
8 $\Pi_m(t) = \sum_{j \in \mathfrak{A}_m} \pi_j(t)$;
9 **if** $\sum_{j \in \mathfrak{A}_m} \theta_j(t) \geq \Delta_m$ **or** $\sum_{j \in \mathfrak{A}_m} \theta_j(t) \geq \Omega_m$ **then**
10 **if** $\Pi_m(t) > \Pi_{m'}(t)$ **then**
11 $t \leftarrow t^*$;
12 $\Delta_m \leftarrow \sum_{j \in \mathfrak{A}_m} \delta_j(t^*)$;
13 $\Omega_m \leftarrow \sum_{j \in \mathfrak{A}_m} \omega_j(t^*)$;
14 $\theta_j(t) \leftarrow 0 \, \forall j \in \mathfrak{A}_{m'}$;
15 **end**
16 **end**
17 **end**
18 **end**

An example of the O-TSCA implemented at an intersection within the microscopic traffic simulation modelling framework described in §4.1 is depicted in Figure 6.6. The line labelled Δ_1 corresponds to the combined demand of all the lanes served during phase 1 (*i.e.* all those lanes running in a west to east and east to west direction) at time t^* when service to phase 1 commenced. The line labelled Ω_1 corresponds to the combined availability of all the lanes served during phase 1 at time t^* when service to phase 1 commenced. Finally, the line labelled $\sum_{j\in\mathfrak{A}_m}\theta_j(t)$ corresponds to the combined throughput of all the lanes served during phase 1 at time t. It may be observed that this line falls to zero soon after it has reached Δ_1. This corresponds to a switch between phases once the initial demand of the phase receiving service has been met, at which point in time the combined throughput of all lanes that were receiving service is reset to zero.

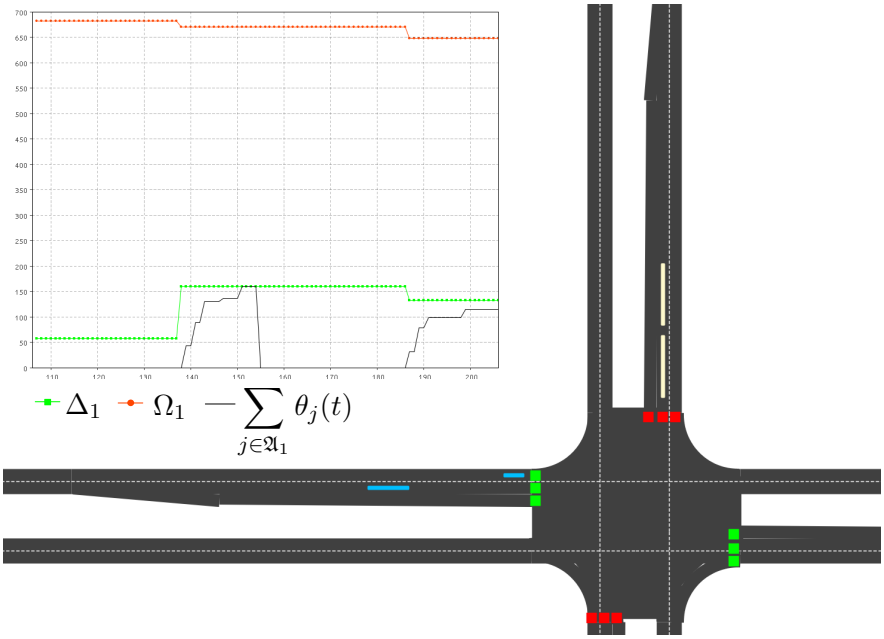


FIGURE 6.6: *An example of the O-TSCA implemented at an intersection within the microscopic traffic simulation modelling framework described in §4.1.*

6.3 A hybrid self-organising traffic signal control algorithm

Initial testing of both the I-TSCA and the O-TSCA revealed interesting findings. It was found that under lighter prevailing traffic demands the I-TSCA outperformed the O-TSCA, while under heavier prevailing traffic demands, the O-TSCA outperformed the I-TSCA in terms of minimising average vehicle delay. The reason for this was that under lighter traffic demand, the O-TSCA provided green times that were longer than necessary, allowing for unnecessary intersection underutilisation. On the other hand, during heavier traffic demand, the I-TSCA switched too frequently between phases, resulting in green times that were too short to adequately clear queues of arriving vehicles. A hybrid traffic signal control algorithm was therefore conceived which combines the I-TSCA and the O-TSCA so as to exploit the best attributes of both algorithms.

The hybrid algorithm (HYBRID) runs both the I-TSCA and the O-TSCA concurrently together with an *intersection utilisation maximisation supervisory mechanism* (IUMSM). The IUMSM

ensures that the intersection for which the traffic control algorithm is responsible is not underutilised due to green times that are too long as well as that service is not terminated prematurely to a platoon of vehicles ready to cross the intersection. It achieves this by considering the proximity of the nearest vehicles to the intersection along all intersection approach lanes. A *proximity variable* $\xi_j(t)$ is associated with approach lane j. If there is a vehicle within its safe following distance of the intersection at time t, then $\xi_j(t) = 1$; else $\xi_j(t) = 0$. A proximity variable $\Xi_m(t) = \sum_{j \in \mathfrak{A}_m} \xi_j(t)$ is associated with phase m. If a request is made by the I-TSCA for a signal change, the IUMSM determines whether or not there is a vehicle within a distance equal to the respective vehicle's safe following distance of the intersection along any approach lane currently receiving service. If there is no vehicle within this specified range of the intersection (*i.e.* $\Xi_m(t) = 0$), then the signal change takes place. If, however, there is a vehicle within this specified range (*i.e.* $\Xi_m(t) = 1$), service is continued until there are no longer any vehicles within the specified range of the intersection along any approach lanes currently receiving service, or until the O-TSCA issues a request for a signal change. This ensures that green times are not terminated prematurely. Analogously, if there is at least one vehicle travelling along an intersection approach lane which is not currently receiving service, and this vehicle will reach the intersection within a time equal to that of an amber and all-red phase, and there is no vehicle within an amber and all-red phase along any approach lane currently receiving service, the IUMSM will issue a change in service if neither the the I-TSCA or O-TSCA has done so yet. This ensures that intersection underutilisation is avoided.

A pseudo code description of HYBRID is presented in Algorithm 6.3. At each time step, the algorithm runs both the I-TSCA and the O-TSCA concurrently, as described in Algorithms 6.1 and 6.2, respectively. If a request is made by the I-TSCA for a signal change, the algorithm determines whether or not there is a vehicle within a distance equal to the respective vehicle's safe following distance of the intersection along any approach lane currently receiving service (line 26). If there is no vehicle within this specified range of the intersection (*i.e.* $\Xi_m(t) = 0$), then the signal change takes place. If, however, there is a vehicle within this specified range (*i.e.* $\Xi_m(t) = 1$), service is continued until there are no longer any vehicles within the specified range of the intersection along any approach lanes currently receiving service, or until the O-TSCA issues a request for a signal change (line 31).

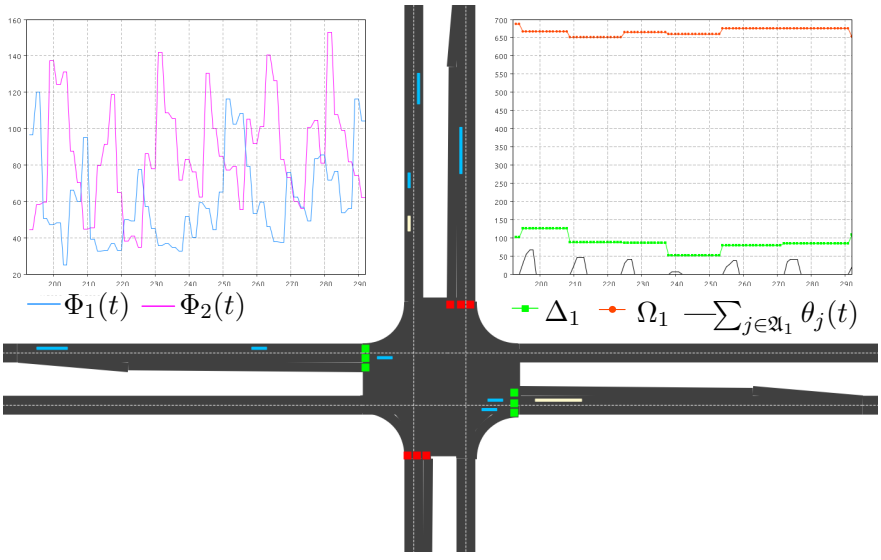


FIGURE 6.7: *An example of HYBRID implemented at an intersection within the microscopic traffic simulation modelling framework described in §4.1.*

Algorithm 6.3: The inventory traffic signal control algorithm

Input : The positions, speeds and physical lengths of all vehicles approaching the intersection at each time step.

Output: A signal phase switching decision for each time step.

for *each time step* t **do**
  for *each approach lane* j *of the intersection* **do**
    $\delta_j(t) \leftarrow \sum_{i \in \mathfrak{C}_j(t)} \hat{\ell}_i$;
    $\omega_j(t) \leftarrow \alpha_{j'} - \delta_{j'}(t)$;
    $\pi_j(t) \leftarrow \delta_j(t) + \omega_j(t)$; **if** $|\mathfrak{Q}_j(t)| = 0$ **and** $|\mathfrak{C}_j(t)| > 0$ **then**
      $\gamma_j(t) = d_{1,\rho_j(t)}(t)/v_1(t)$;
    end
    else if $0 < |\mathfrak{Q}_j(t)| < \infty$ *and* $|\mathfrak{C}_j(t)| > 0$ **then**
      $\gamma_j(t) = |\mathfrak{Q}_j(t)|/\eta_j$;
    end
    else if $|\mathfrak{C}_j(t)| = 0$ **then**
      $\gamma_j(t) = \infty$;
    end
    for *each vehicle* i *on approach lane* j **do**
      if $\kappa_j(t) = 1$ **then**
        $\phi_{ij}(t) = \tau_m(t) + \frac{\epsilon_i(t)}{\eta_j} - \frac{d_{i,\mu_i}(t)}{v_i(t)}$;
      end
      else if $\kappa_j(t) = 0$ **then**
        $\phi_{ij}(t) = \sum_{p \in \mathfrak{P}} \tau_p(t) + \sum_{p \in \mathfrak{P} \setminus \{m\}} \chi_p(t) - \frac{d_{i,\mu_i}(t)}{v_i(t)} + \frac{\epsilon_i(t)}{\eta_j}$;
      end
    end
  end
  for *each signal phase* m **do**
    $\Gamma_m(t) = \min_{j \in \mathfrak{A}_m} \gamma_j(t)$;
    $\Phi_m(t) = \sum_{j \in \mathfrak{J}} \sum_{i \in \mathfrak{C}_j(t)} \phi_{ij}(t)$;
    if $\Phi_m(t) < \Phi_{m'}(t)$ **and** $\chi_m(t) = 0$ **and** $\Xi_m(t) = 0$ **then**
      $\kappa_j(t) \leftarrow 1 \forall j \in \mathfrak{A}_m$;
      $\chi_m(t) \leftarrow \Gamma_m(t)$;
    end
    $\Pi_m(t) = \sum_{j \in \mathfrak{A}_m} \pi_j(t)$;
    if $\sum_{j \in \mathfrak{A}_m} \theta_j(t) \geq \Delta_m$ **or** $\sum_{j \in \mathfrak{A}_m} \theta_j(t) \geq \Omega_m$ **or** $(\Xi_m(t) \geq 1$ **and** $\Xi_{m'}(t) = 0)$ **then**
      if $\Pi_m(t) > \Pi_{m'}(t)$ **then**
        $t \leftarrow t^*$;
        $\Delta_m \leftarrow \sum_{j \in \mathfrak{A}_m} \delta_j(t^*)$;
        $\Omega_m \leftarrow \sum_{j \in \mathfrak{A}_m} \omega_j(t^*)$;
        $\theta_j(t) \leftarrow 0 \forall j \in \mathfrak{A}_{m'}$;
      end
    end
  end
end

An example of HYBRID implemented at an intersection within the microscopic traffic simulation modelling framework described in §4.1 is depicted in Figure 6.7. It may be observed that in the scenario depicted, the line labelled $\sum_{j\in\mathfrak{A}_m}\theta_j(t)$ does not ever reach the line labelled Δ_1. This stands in contrast to Figure 6.6. The reason for this is that either I-TSCA or the IUMSM issues a change in signal before all the vehicles detected by O-TSCA along roadways receiving service during phase 1 have passed through the intersection. This is indicative of shorter, more frequent green-time phases.

6.4 Chapter summary

In this chapter three novel self-organising traffic signal control algorithms were introduced and described. All three algorithms assume the use of radar detection sensors similar to those described in §1.2 . The first traffic signal control algorithm, presented in §6.1, was inspired by inventory control theory and draws parallels between the monetary costs involved in inventory control and the delay time costs associated with traffic control at signalised intersections. The second traffic signal control algorithm, presented in §6.2, was inspired by the chemical process of osmosis. The third traffic signal control algorithm, presented in §6.3, is a hybrid algorithm which combines the previous two algorithms and supplemented them with an intersection utilisation maximisation technique.

CHAPTER 7

Algorithmic comparison and evaluation

Contents

This chapter opens with a description in §7.1 of the experimental design adopted and the various scenarios and conditions under which the traffic control algorithms of Chapters 5 and 6 were tested. In §7.1.2, the general parameters and settings applied during each simulation run are described, while in §7.1.3, the values of the parameters specific to each traffic control algorithm are described for the various prevailing traffic scenarios. In §7.1.4, the performance measures which are used to evaluate and compare the efficacies of the various traffic control algorithms are discussed. The results obtained are presented in §7.2 . The Tukey Honest Significant Difference method is used to rank the algorithms and is described briefly in §7.2.1 . The results obtained for a corridor comprising four homogeneous intersections are presented in §7.2.2, while the results obtained for a three-by-four grid of twelve homogeneous intersections are presented in§7.2.3. A final appraisal of each of the traffic signal control algorithms investigated is presented in §7.3. The chapter closes with a brief summary in §7.4.

7.1 Experimental design

Six traffic signal control algorithms are compared and evaluated in this chapter according to the performance measures described in §4.2. The algorithms are the traffic signal control algorithms of Lämmer and Helbing (henceforth referred to as *LH*), Gershenson (henceforth referred to as *Gersh*), the novel traffic signal control algorithm inspired by inventory theory (henceforth referred to as I-TSCA), the novel traffic signal control algorithm inspired by the process of

osmosis (henceforth referred to as O-TSCA), the novel hybrid traffic signal control algorithm (henceforth referred to as *Hybrid*), and a fixed-time traffic signal control algorithm (henceforth referred to as *Fixed*), which combines the fixed-time control approach described in §5.2.1 with the green-wave method described in §5.2.2. Thus, for Fixed, the cycle length and green times at each intersection are calculated according to the procedure for equalising the degree of saturation on all intersection approaches, as proposed in the Highway Capacity Manual [94], while the offset between green signals of adjacent intersections is calculated according to the green-wave method proposed in [38].

In this dissertation, the various traffic signal control algorithms are implemented and compared for two distinct road network topologies. The first topology is a corridor comprising four intersections, while the second is a three-by-four grid of intersections. For each topology, two different scenarios are investigated, pertaining to the arrival rate at which vehicles enter the road network. In the first scenario, the average vehicle arrival rate remains constant throughout the simulation, while in the second scenario, the average vehicle arrival rate varies according to a function of the time elapsed since the start of the simulation run.

In the first scenario, each algorithm is tested under three prevailing traffic density conditions: light, medium and heavy traffic flow conditions [94]. In the light traffic flow scenario, the mean arrival rate for each entry approach to the traffic network is taken as 10 vehicles per minute. In the medium and heavy traffic flow scenarios, these mean arrival rates were taken as 20 vehicles per minute and 30 vehicles per minute, respectively.

For each traffic flow scenario and traffic control algorithm combination, the simulation model is run thirty times. Each simulation run is the equivalent of one hour of simulated traffic flow, preceded by a sufficiently large warm-up period.

The method according to which the length of the warm-up period is calculated is described in §7.1.1 after which the general traffic control algorithm conditions and parameter settings for each road network topology and vehicle arrival rate scenario are described in §7.1.2. This is followed by a discussion on the specific parameter settings (if any) of each of the algorithms implemented for the various scenarios considered. The performance measures in terms of which the algorithms are evaluated are finally described in §7.1.4.

7.1.1 The length of the warm-up period

When investigating certain simulation model performance measures over an extended period of time, it is often the case that the initial few observations do not provide a true representation of the steady-state behaviour of the model. To account for this lack of representation, a warm-up period is usually introduced into the simulation model during which all observations made are ignored or discarded [27]. A natural question arising from the introduction of a warm-up period is, however, how long the warm-up period should be. The method proposed by Law in [56] is employed to determine the lengths of the warm-up periods.

When attempting to estimate the steady-state mean $\bar{m} = E(Y)$, of a system, which is also generally defined by

$$\bar{m} = \lim_{i \to \infty} E(Y_i),$$

where $Y_1, Y_2, Y_3, \ldots$ are observations of a variable Y in a simulation model, it may be seen that the transient means converge to the steady state mean. However, Law [56] describes the *problem of the initial transient* where $E[\bar{Y}(x)] \neq \bar{m}$ for any finite period of time x. To overcome this problem, he suggests that a warm-up period of length x^* should be introduced during which all

observations are ignored, and that

$$\bar{Y}(x, x^*) = \frac{\sum_{i=x^*+1}^{x} Y_i}{x - x^*}$$

should be used as an estimator of $\bar{m}$ rather than $\bar{Y}(x)$, for all $1 \leq x^* \leq x - x^*$. To determine the value of x^*, Law [56] suggests employing the following four steps:

1. Produce w replications of the simulation, each of length x, letting Y_{ij} be the i^{th} observation of the variable Y from the j^{th} replication, for all $i = 1, 2, \ldots, x$ and $j = 1, 2, \ldots, w$.

2. Let $\bar{Y}_i = \sum_{j=1}^{w} Y_{ij}/w$ for $i = 1, 2, \ldots, x$. Thus, the averaged processes $\bar{Y}_1, \bar{Y}_2, \ldots, \bar{Y}_x$ have means $E(\bar{Y}_i) = E(Y_i)$ and variances $\text{Var}(\bar{Y}_i) = \text{Var}(Y_i)/w$, *i.e.* the averaged process has the same transient mean curve as the original process, but its plot has only $(1/w)$-th of the variance.

3. To smooth out the high frequency oscillations in $\bar{Y}_1, \bar{Y}_2, \ldots, \bar{Y}_x$ while at the same time leaving the low-frequency oscillations or trends which are of interest to the investigation, a moving average

$$\bar{Y}_i(y) = \begin{cases} \frac{\sum_{s=-y}^{y} \bar{Y}_{i+s}}{2y+1} & \text{if } i = y+1, \ldots, x-y \\ \frac{\sum_{s=-(i-1)}^{i-1} \bar{Y}_{i+s}}{2i-1} & \text{if } i = 1, \ldots, y \end{cases}$$

   is computed, where y is the window of the moving average and is a positive integer such that $y \leq \lfloor x/4 \rfloor$.

4. A value of x^* is chosen as i beyond which $\bar{Y}_i(y), \bar{Y}_{i+1}(y), \ldots, \bar{Y}_{x-y}(y)$ appears to have converged.

In the case of the simulation models implemented in this chapter, Y_{ij} is taken as the total number of vehicles present in the system at observation point i during simulation run j. Each simulation was run for the equivalent of 2 400 seconds (or 40 minutes), unless no steady state was apparent after this time, in which case this time was extended to allow for the emergence of a steady state. Observations were made every 10 seconds (*i.e.* $x = 240$). The number w of replications produced for each arrival rate was initially set to 10. This value was, however, increased if it did not yield a satisfactory indication of a steady state, as prescribed by Law [56]. For the cases of low and medium traffic flows, a sufficient warm-up period was found to be the equivalent of 30 minutes, as opposed to 15 minutes for the case of heavy traffic flow.

For every simulation run, regardless of road network topology or arrival rate, a (pseudo) random number generator is used to generate travel plans for the vehicles during each simulation (including the arrival times of vehicles, their destinations, their desired speeds, *etc*). While the seed of this random number generator is different for each of the 30 simulation runs, the same 30 seeds are used when testing of each algorithm in order to ensure a fair comparison between the algorithms tested.

In the scenario where the average vehicle arrival rate varies as a function of time, the average arrival rate is varied every minute, following a warm-up period. During the warm-up period, the average arrival rate remains fixed at 10 vehicles per minute at each road network entry point. This warm-up period is again set to the equivalent of 30 minutes. Following this warm-up the simulation is run for an equivalent of one hour and forty minutes (or 100 minutes) during which the average arrival rate is drawn from a discrete uniform distribution every minute, on the

minute. The parameters of the uniform distribution are adjusted every twenty-minutes. For the first twenty-minute period (following the warm-up period), the arrival rate drawn each minute is uniformly distributed over the interval of zero to twenty vehicles per minute, and therefore has an expected value of ten vehicles per minute. For the following twenty-minute period the arrival rate is uniformly distributed between ten vehicles per minute and thirty vehicles per minute (with an expected value of twenty vehicles per minute). For the third twenty-minute period, the average arrival rate is uniformly distributed between twenty vehicles per minute and forty vehicles per minute (with an expected value of thirty vehicles per minute). The arrival rates for the fourth and fifth twenty-minute periods are drawn from the same distributions as those for the first and second twenty-minute periods, respectively.

7.1.2 General model conditions and parameters

Uniform, homogeneous intersections are assumed in this chapter. Each intersection comprises four approach roads, consisting of three lanes each at the intersection, as may be seen in Figure 7.2. Vehicles may turn left or travel straight through the intersection from the left-most lane of each approach, or travel straight through the intersection from the middle lane, or turn right at the intersection from the right-most lane (which is an exclusive right-turn lane). One of four traffic control phases may be implemented at any one time at each intersection. These four phases are shown in Figure 7.2.

Approaches either receive an *all-green* phase or an *exclusive right-turn* phase. During an all-green phase vehicles travelling along an approach receiving a green signal may enter the intersection and all vehicles wishing to turn right, do so on a *permitted* basis (*i.e.* they are required to wait in the intersection for a sufficiently large gap in both space and time in oncoming traffic before proceeding through the intersection). Between every two consecutive phases, a mandatory setup phase is implemented, consisting of amber and all-red periods which are set to 2 seconds and 3 seconds in duration, respectively. All vehicles travelling from west to east or from east to west are served during the same phase and, analogously, all vehicles travelling from north to south or from south to north are served during the same phase.

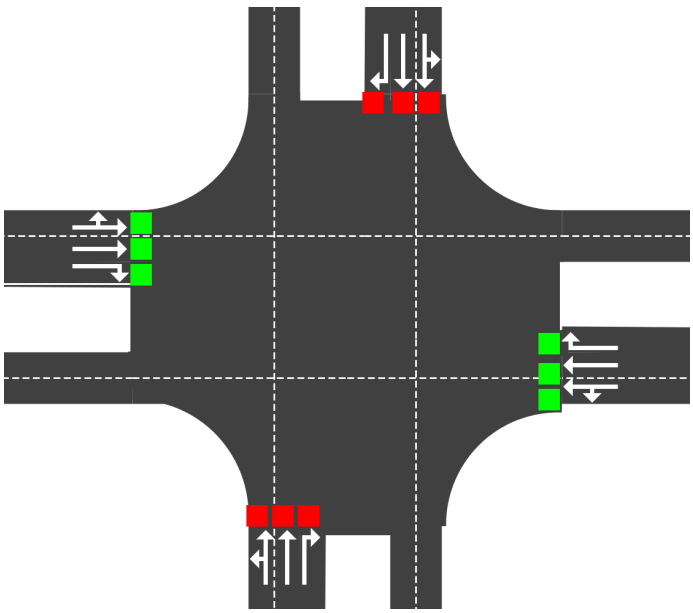


FIGURE 7.1: *Generic intersection design.*

Vehicles are generated at each entry point of the traffic network according to a Poisson distribution with a mean of λ vehicles per minute.[1] Upon generation, vehicles are assigned a length, a desired travelling speed and a destination. Vehicles may be 5 metres, 10 metres or 15 metres in length (representing light motor vehicles, small heavy motor vehicles and large heavy motor vehicles, respectively [72]). The length selection is random and generated according to a Monte Carlo simulation process, for which there is a 10% probability that a vehicle with a length of 15 metres will be generated, a 30% probability that a vehicle with a length of 10 metres will be generated and a 60% chance that a vehicle with a length of 5 metres will be generated.

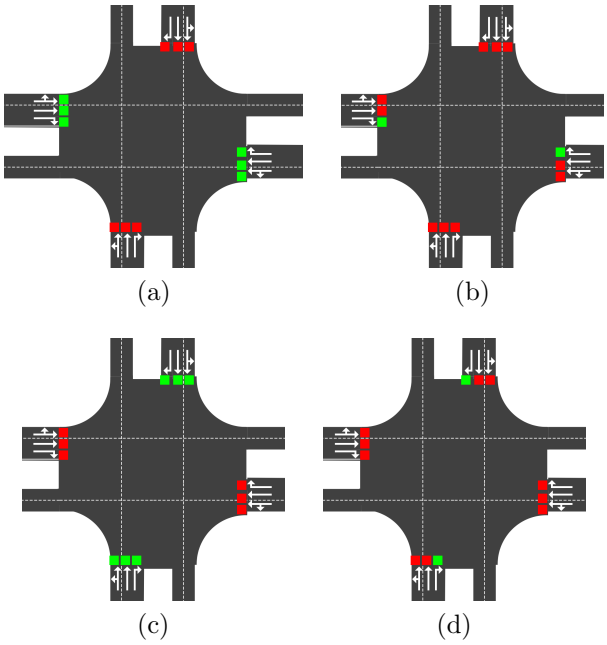


FIGURE 7.2: *Four traffic control phases. (a) All vehicles travelling from west to east or from east to west receive a green signal. Vehicles turning right, do so on a permissive basis. (b) Exclusive right-turn phase for vehicles travelling from west to east or from east to west. (c) All vehicles travelling from north to south or from south to north receive a green signal. Vehicles turning right, do so on a permissive basis. (d) Exclusive right-turn phase for vehicles travelling from north to south or from south to north.*

The speed limit of the model is set to 60 kilometres per hour (or 17 metres per second, approximately). The desired speed of each vehicle generated is calculated by multiplying the speed limit by a *speeding factor*. This speeding factor is a continuous random value which is uniformly distributed between 0.7 and 1.2. The functions described in §4.1.3 pertaining to the vehicle-following characteristics of the model are defined in Table 7.1. The functions in the table are the pre-existing functions implemented by the road traffic library of AnyLogic. The combination of this speed limit together with the vehicle-following functions described result in an expected average maximum vehicle flow rate of approximately 1.2 vehicles per second, or 72 vehicles per hour.

The origin-destination pairings of the vehicles are determined in such a manner that a vehicle need only travel straight along a single corridor, or turn left or right at most once to reach its destination. An example of the possible origin-destination pairings of a vehicle entering the three-by-four grid road network topology at the most north-westerly point and travelling from west to east initially is shown in Figure 7.3. Following this origin-destination pairing methodology, a vehicle entering the road network (be it the corridor or grid topology) travelling

[1] This is equivalent to exponentially distributed vehicle interarrival times with a mean of $1/\lambda$ minutes per vehicle.

Description	Function
The minimum allowable following distance of vehicle i (metres).	$f(v_i) = v_i/2 + 1$
The maximum allowable following distance of vehicle i (metres).	$f'(v_i) = (v_i/2 + 1) \times 1.5$
The maximum speed of vehicle i while following vehicle $i-1$ (metres per second).	$g(s_{i,i-1}) = (s_{i,i-1} - 1) \times 2$
The minimum speed of vehicle i while following vehicle $i-1$ (metres per second).	$g'(s_{i,i-1}) = (s_{i,i-1}/1.5 - 1) \times 2$
The maximum speed of vehicle i on a curved road with radius x_{radius} (metres per second).	$h(x_{\text{radius}}) = x_{\text{radius}}/2$

Table 7.1: *The characteristic car-following functions implemented for the simulation study, as described in §4.1.3.*

in a west-to-east or an east-to-west direction may select one of nine possible destinations. For such a vehicle, the turning probabilities are assumed as follows: A vehicle travels straight along a single corridor with a 60% probability and turns left or right at any one of the four intersection it encounters on its original bearing with a probability of 5% each. On the other hand, for the case in which a vehicle enters the road network travelling in a north-to-south or a south-to-north direction, the number of possible origin-destination pairings varies depending on whether the corridor or grid topology is implemented. In the former case, the vehicle may select one of three origin-destination pairings. The vehicle will travel straight with a probability of 90% or it may turn left or right at the first intersection it encounters with a probability of 5% each. In the latter case, in which the grid topology is implemented, there are seven possible origin-destination routes the vehicle may follow. In such an instance the vehicle travels straight with a probability of 70%, or it turns left or right at any one of the three intersections it encounters on its original bearing with a probability of 5% each. To summarise, a vehicle will travel straight by default, otherwise it will turn left or right with a probability of 5% each at any intersection it encounters along its original bearing.

7.1.3 Traffic signal control parameter settings

For the fixed-time traffic signal control algorithm, the cycle time and green times are calculated according to (5.4) and (5.5), respectively, while the offsets between adjacent intersections are calculated according to (5.7). For the cases where the average arrival rate remains fixed at 10 vehicles per hour, twenty vehicles per hour or thirty vehicles per hour, the cycle times implemented at each intersection are 14 seconds, 24 seconds or 70 seconds, respectively, while the green times implemented for each phase are 2 seconds, 7 seconds or 30 seconds, respectively (no exclusive right-turn phases are implemented). The offset time is taken as the time it would take a vehicle to travel between two intersections at the speed limit. For the scenario in which the corridor road network topology is implemented, the intersection offsets are implemented such that the green wave moves in a west-to-east direction. For the scenario in which the grid road network topology is implemented, the green waves move in a west-to-east direction and in a north-to-south direction. This is achieved by switching all the traffic signals which lie on the same diagonal running from south west to north east at the same time.

For Gersh, the parameter values are taken as recommended in [38]. The threshold value ς and minimum green time value u are taken as 13.33 seconds and 3.33 seconds, respectively, while

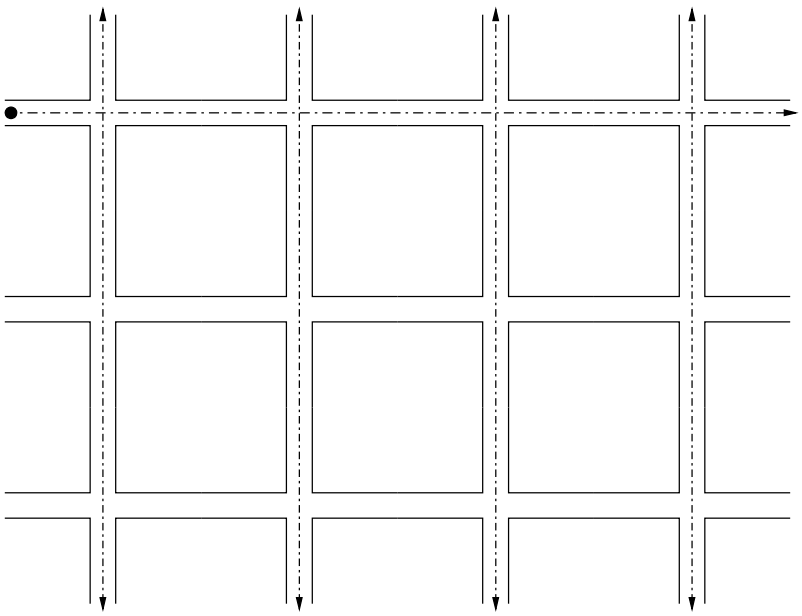


FIGURE 7.3: *The possible origin-destination pairings, and the routes connecting them, of a vehicle entering the road network at the most north-westerly entry point, and travelling from west to east.*

the values of d, r and e are taken as 50 metres, 25 metres and 10 metres, respectively. Finally, the maximum number of vehicles s which can prolong a green signal is taken as two vehicles.

The parameters of LH are taken as recommended in the example in [54]. The value of Z is taken as 90 seconds, while the value of $Z^{\max}$ is taken as 120 seconds.

None of I-TSCA, O-TSCA or Hybrid requires any explicit settings of parameters.

7.1.4 Performance measure indicators

The output capabilities of the traffic simulation modelling framework were described in in §4.2. From this output, the performance measure indicators that were selected to compare the various traffic control algorithms include the mean and maximum of the delay time and normalised delay time experienced by vehicles, the mean number of stops as well as the normalised number of stops made by vehicles, and the saturation of the system. These same performance indicators are also generated for all vehicles travelling in a west-to-east direction or in an east-to-west direction as well as for all vehicles travelling in a north-to-south direction or in a south-to-north direction. The performance measure indicators are supplemented with additional output data, such as the green times implemented for each phase at each intersection, in order to aid in the analysis of the results generated.

7.2 Simulation results and analyses

In this section, results are presented in the form of box-and-whisker plots for each of the six traffic control algorithms tested with respect to the mean and maximum delay time and normalised delay time experienced as well as the mean number of stops made and mean normalised number of stops made for light, medium and heavy prevailing traffic flow conditions. In addition, for the scenario in which the vehicle arrival rates vary over time, time plots are presented which

depict how the mean delay time, mean number of stops made and saturation of the roadways vary as a result of the changing arrival rates. Furthermore, preference rankings of the various traffic control algorithms are provided in respect of each performance measure indicator. An analysis and interpretation of the graphical results follows. An *analysis of variance* (ANOVA) test is carried out for each set of results in order to determine whether or not there was a significant difference in the performances of the various algorithms (at a 95% confidence level). This is followed by an application of the Tukey *Honest Significant Difference* (HSD) method [4] in order to determine which algorithms differed significantly from each other in terms of the various performance measures and by what margin (again at a 95% confidence level), thus, effectively ranking the algorithms in a statistically significant manner.

7.2.1 The Tukey Honest Significant Difference method

The Tukey HSD method compares the expected performance p_i ($i \in \{1, \ldots, a\}$) of a different simulated systems in terms of a selected performance measure and ranks the a different systems accordingly, by forming simultaneous confidence intervals in respect of the parameters $p_i - p_\ell$ for all $i \neq \ell$. These $a(a-1)/2$ confidence intervals indicate the magnitudes and directions of the differences between the performances of each pair of alternative systems. Supposing that the systems are simulated independently to obtain independent and identically distributed normal output data $Y_{i1}, Y_{i2}, \ldots, Y_{iw_i}$ from system i, the sample mean of system i is

$$\bar{Y}_i = \sum_{j=1}^{w_i} Y_i / w_i$$

and the pooled sample variance of all a systems is

$$S^2 = \frac{1}{a} \sum_{i=1}^{a} \frac{1}{w_i - 1} \sum_{j=1}^{w_i} \left(Y_{ij} - \bar{Y}_i\right)^2 = \frac{1}{a} \sum_{i=1}^{a} \frac{1}{w_i - 1} \left(\sum_{j=1}^{w_i} Y_{ij}^2 - \bar{Y}_i^2 \right).$$

The simultaneous confidence intervals for $p_i - p_\ell$ are then calculated as

$$\bar{Y}_i - \bar{Y}_l \pm \frac{Q_{a,\nu}^{(\alpha)}}{\sqrt{2}} S \sqrt{\frac{1}{w_i} + \frac{1}{w_\ell}}$$

for all $i \neq \ell$, where $Q_{a,\nu}^{(\alpha)}$ is the $1-\alpha$ quantile of the Studentised range distribution with parameter a and $\nu = \sum_{i=1}^{a}(w_i - 1)$ degrees of freedom. If $w_1 = w_2 = \cdots = w_a$, then these confidence intervals achieve a simultaneous coverage probability of $1 - \alpha$.

For the comparisons made in this chapter, a is the number of different traffic control algorithms tested (*i.e.* six), w_i is the number of iterations implemented of each traffic control algorithm (*i.e.* thirty) and $\alpha = 0.05$ since a 95% confidence level is assumed, which results in $\nu = 174$ degrees of freedom.

7.2.2 A traffic corridor comprising four homogeneous intersections

In this section, results and analyses are presented pertaining to the corridor road network topology, first for fixed constant vehicle arrival rates and then for varying vehicle arrival rates. Figure 7.4 depicts the road traffic corridor implemented in the microscopic traffic simulation modelling framework described in §4.2.

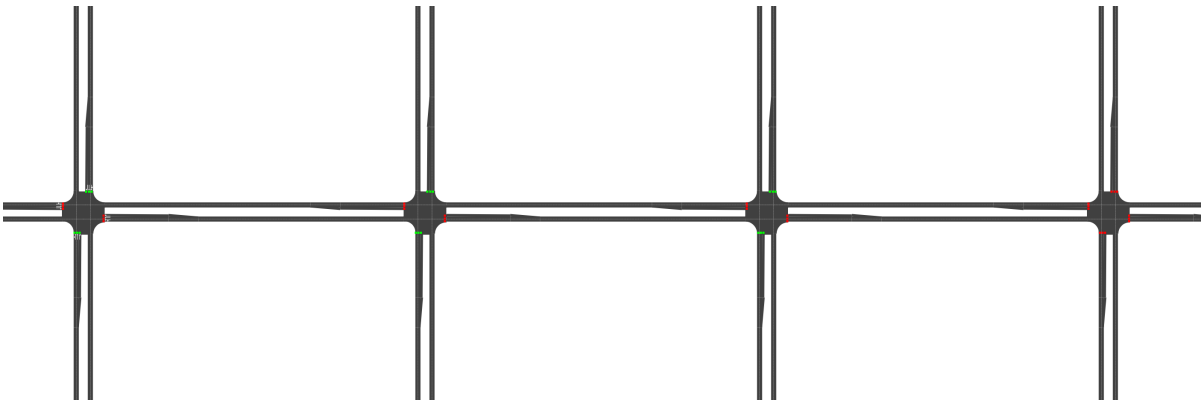


FIGURE 7.4: *The road traffic corridor implemented in §7.2.2 within the microscopic traffic simulation modelling framework described in §4.2.*

Simulation results for light traffic flow conditions ($\lambda = 10$ vehicles per minute)

For the performance measure of mean delay time, it was found that there was a significant difference between at least two of the traffic control algorithms under light traffic flow conditions, suggesting that the null hypothesis (that all the traffic control algorithms are equivalent) may be rejected. Hybrid was found to be the best performing algorithm (with an average mean delay time of 11.89 seconds), while the worst performing algorithm was Fixed (with an average mean delay time of 17.48 seconds) as may be seen in Figure 7.5(a). The rankings of the six different traffic control algorithms as well as their performances in relation to one another in reference to average delay time experienced for the scenario in which $\lambda = 10$ are summarised in Table 7.2. Similar tables for the various other arrival rates and performance measure indicators may be found in Appendix B. At a 95% confidence level, Hybrid significantly outperformed Gersh, LH and Fixed by 6.2%, 26.7% and 32%, respectively. Hybrid also achieved the lowest average maximum delay time of 89.02 seconds, as indicated in Figure 7.5(e), although this was not significantly better than that of Gersh or I-TSCA. Hybrid did, however, achieve the lowest maximum delay time experienced by a vehicle, namely 99 seconds. The results for the performance measure of the mean normalised delay time, shown in Figure 7.5(c), follow much the same pattern as those for the mean delay time. Hybrid again significantly outperformed the other five algorithms, with an average normalised delay time of 1.18, indicating that the travel times of vehicles in the system were 18% times more than their ideal travel times.

When considering the mean number of stops, it was found that O-TSCA was the best performing algorithm with a value of 0.87 stops, followed by LH and Hybrid, which it significantly outperformed by 7.5% and 12.8%, respectively, as shown in Figure 7.5(b). On the other hand, LH significantly outperformed all five of the other algorithms, including O-TSCA, with respect to the mean normalised number of stops made, as shown in Figure 7.5(d). A reason for this is that although O-TSCA resulted in fewer stops being made, on average, by vehicles travelling from west to east or from east to west, when compared to LH (1.7 compared to 2.3, respectively), LH resulted in fewer stops being made by vehicles travelling from north to south or from south to north when compared to O-TSCA (0.6 compared to 0.66, respectively). This proved to be significant as, on average, approximately 4 800 vehicles travelled from north to south or from south to north, compared to approximately 1 200 vehicles on average, which travel from west to east or from east to west. Thus, while reducing the number of stops made by vehicles travelling along the corridor reduces the overall number of stops made, reducing the number of stops made by vehicles travelling across the corridor reduces the normalised number of stops made.

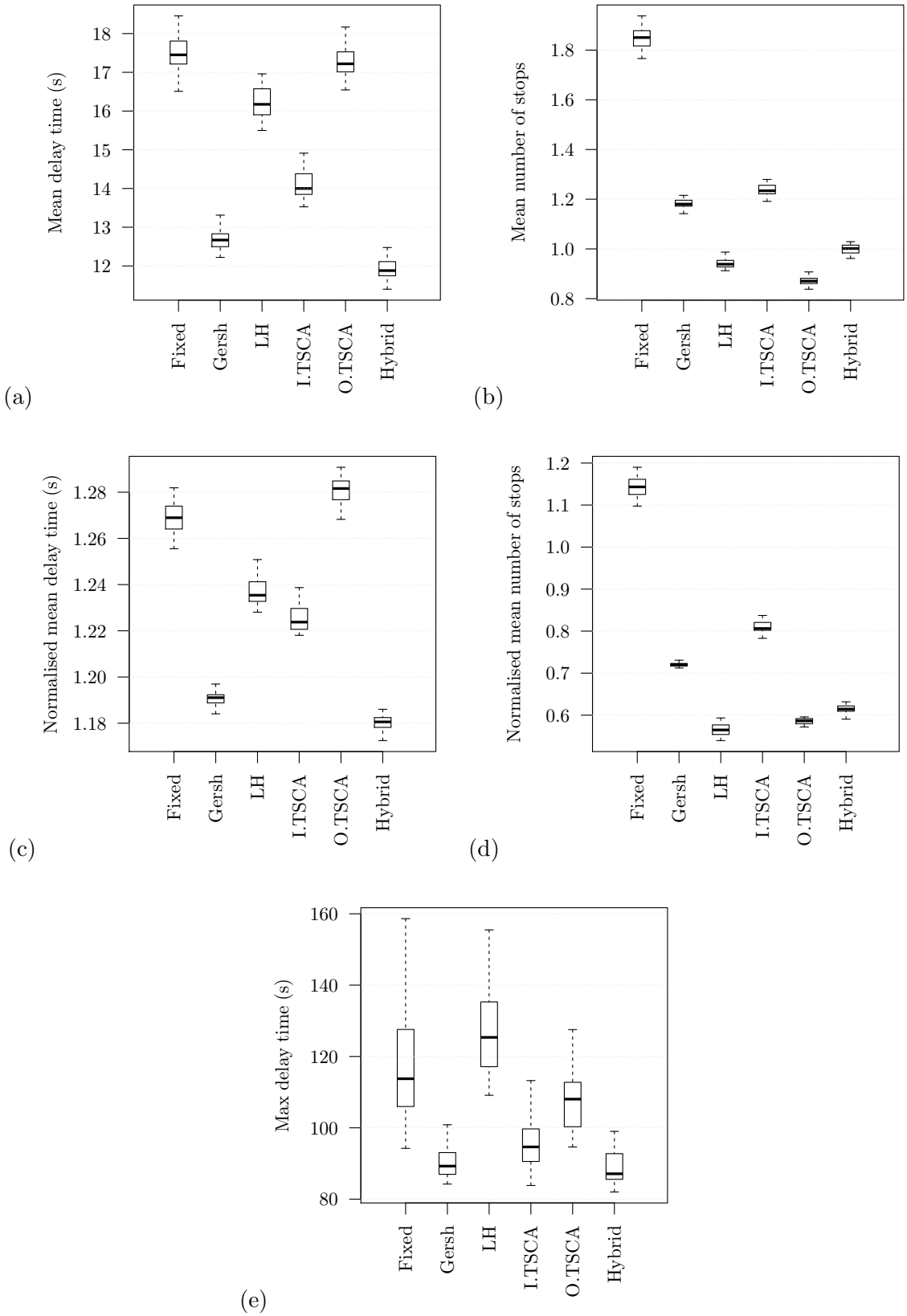


FIGURE 7.5: *Results for $\lambda = 10$ vehicles per minute for the corridor road network topology. (a) Mean delay time. (b) Mean number of stops. (c) Mean normalised delay time. (d) Mean normalised number of stops. (e) Maximum delay time.*

Traffic control algorithm 1	2	Average Improvement (s)	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
Hybrid	Gersh	0.78	6.2%	0.52	1.04	4.1%	8.2 %
Hybrid	I-TSCA	2.19	15.6%	1.93	2.45	13.7%	17.4%
Hybrid	LH	4.33	26.7%	4.07	4.59	25.1%	28.3%
Hybrid	O-TSCA	5.36	31.1%	5.10	5.62	29.6%	32.6%
Hybrid	Fixed	5.59	32.0%	5.33	5.85	30.5%	33.5%
Gersh	I-TSCA	1.41	10.0%	1.15	1.67	8.2%	11.9%
Gersh	LH	3.55	21.9%	3.29	3.81	20.3%	23.5%
Gersh	O-TSCA	4.58	26.5%	4.32	4.84	25.0%	28.1%
Gersh	Fixed	4.81	27.5%	4.55	5.07	26.0%	29.0%
I-TSCA	LH	2.14	13.2%	1.88	2.40	11.6%	14.8%
I-TSCA	O-TSCA	3.17	18.4%	2.91	3.43	16.9%	19.9%
I-TSCA	Fixed	3.40	19.4%	3.14	3.66	17.9%	20.9%
LH	O-TSCA	1.03	6.0%	0.77	1.29	4.5%	7.5%
O-TSCA	Fixed	0.23	1.3%	−0.03	0.49	−0.2%	2.8%

Table 7.2: *Algorithmic comparison and ranking with respect to mean delay time for the scenario in which* $\lambda = 10$ *for the corridor road network topology.*

By nature, Gersh, I-TSCA and Hybrid are the more flexible traffic signal control algorithms and, as a result, yielded comparatively short green times, with those of Hybrid being on average approximately 1.4 seconds longer than those of Gersh and I-TSCA, resulting in a decrease in both vehicle delay and the number of stops made by vehicles. However, Fixed, employed the shortest green times of all the algorithms and it performed the worst in four of the five performance measure comparisons. LH and O-TSCA employed considerably longer green times and while these were clearly less effective at reducing vehicle delay, they were very effective at enabling green waves, as indicated by their superior performances at reducing the average number of stops made by vehicles. Overall, it is concluded that Hybrid may be considered to be the best performing algorithm under light traffic flow conditions as it achieves the best balance between flexibility and coordination.

Simulation results for medium traffic flow conditions ($\lambda = 20$ vehicles per minute)

Gersh achieved the lowest average mean delay time of 21 seconds under medium traffic flow conditions, demonstrating a statistically significant improvement of 3% over the next best algorithm, Hybrid. The worst performing algorithm with respect to minimising mean delay time was LH with an average mean delay time of 28.6 seconds, as shown in Figure 7.6(a). Hybrid once again achieved the lowest average maximum delay time of 120.27 seconds, shown in Figure 7.6(e), but this was not significantly better than that of Gersh and O-TSCA. O-TSCA achieved the lowest maximum delay time of 137.92 seconds. Neither I-TSCA nor O-TSCA significantly outperformed each other with respect to mean delay time, but they were both significantly outperformed by Hybrid, illustrating the effectiveness of the IUMSM. The mean normalised delay time results, shown in Figure 7.6(c), are similar to those for the mean delay time. Again it may be seen that Gersh significantly outperformed the other five algorithms, achieving a mean normalised delay time of 1.32. Both I-TSCA and LH outperformed O-TSCA as a result of serving those vehicles travelling from north to south or from south to north either more frequently (in the case of I-TSCA) or for longer periods of time (in the case of LH).

Once again, O-TSCA proved to be superior when considering the mean number of stops made, achieving an average of 1 stop, which was a statistically significant 8.7% improvement over Hybrid, the next best algorithm, as may be seen in Figure 7.6(b). However, there was no statistically significant difference, at a 95% confidence level, between the performances of O-TSCA and Hybrid in respect of the mean normalised number of stops made by vehicles, as shown in Figure 7.6(d). It may be seen again from Figures 7.6(b) and 7.6(d) that while O-TSCA results in fewer stops overall when compared to Hybrid, Hybrid is more adept at equalising service to all approaches of the intersection, thereby reducing the normalised number of stops made. Based on the results in Figure 7.6, Hybrid may again be considered as the best performing algorithm overall for medium traffic flow conditions.

As with light traffic flow conditions, shorter green times resulted in reduced delay times while longer green times resulted in a reduced number of stops in the case of medium traffic flow conditions. However, these discrepancies were considerably less pronounced, with evidence suggesting that slightly longer green times may be beneficial overall, as the average green times implemented by Hybrid (excluding exclusive right turn phases) were almost double those of Gersh, which would appear to be the next best performing algorithm, overall.

Simulation results for heavy traffic flow conditions ($\lambda = 30$ vehicles per minute)

The large variances associated with the performances of Gersh and I-TSCA in the case of heavy traffic flow conditions resulted in a large pooled sample variance and thus large confidence intervals, as calculated by the Tukey HSD method. Therefore, even though Fixed, which achieved an average mean delay time of 32.91 seconds, is shown in Figure 7.7(a) to have outperformed Hybrid, O-TSCA and LH by 4.3%, 9.3% and 17.9%, respectively, these improvements could not be considered statistically significant at a 95% confidence level. Fixed also exhibited the lowest average maximum delay time of 164 seconds, as shown in Figure 7.7(e), but again this was not statistically significant at a 95% confidence level. The discrepancies between the performances of Fixed and Hybrid, Fixed and O-TSCA, and Fixed and LH are relatively smaller when compared in terms of the mean normalised delay times experienced, as shown in Figure 7.7(c), and indicate that the travel time of a vehicle is expected to be between 0.5 and 0.6 times longer than its ideal travel time. This value increases considerably for the case in which Gersh or I-TSCA are implemented during heavy traffic flow conditions.

As for the mean delay time, Fixed was again the best performing algorithm with respect to minimising the mean number of stops made by vehicles, as shown in Figure 7.7(b), with a value of 1.25 stops on average. This is a 2.2% improvement over Hybrid, the next best performing algorithm. Again, this was, however, not statistically significant at a 95% confidence level.

Based on these results, it would appear that Fixed is the best overall performing algorithm for heavy traffic flow conditions, but from a statistical significance point of view, it is interchangeable with both Hybrid and O-TSCA at a 95% confidence level.

The average green times implemented by Gersh were the largest of all six algorithms, while those implemented by I-SOTCA were the smallest. As traffic flow and roadway saturation increases, so the demand along intersection approach lanes tends to become more constant, lending itself towards fixed-time control. This explains the superior performance of Fixed, as well as those of Hybrid and O-SOTCA, which were able to implement average green times which were very similar in duration to those of Fixed, with very little variance.

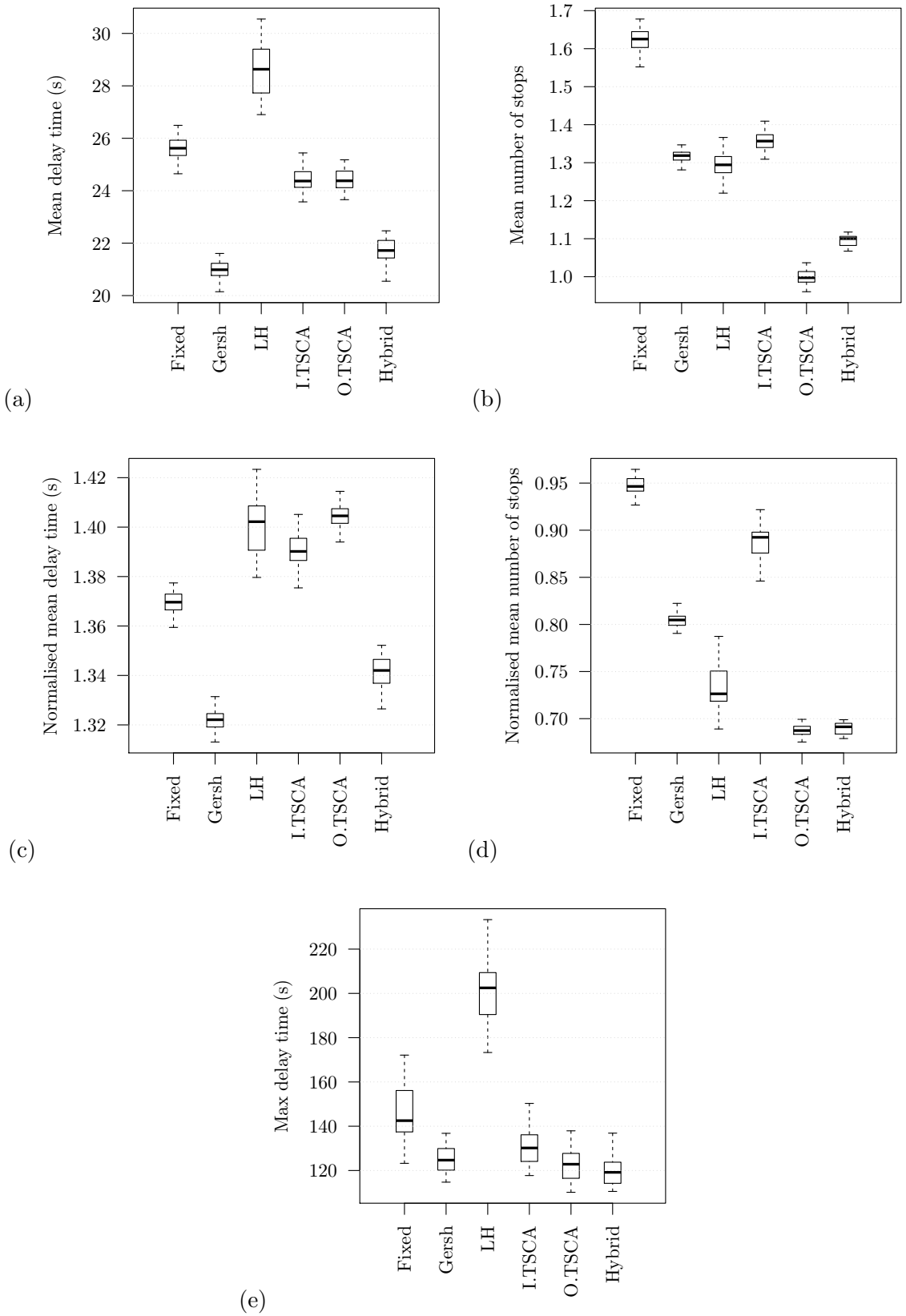


FIGURE 7.6: *Results for $\lambda = 20$ vehicles per minute for the corridor road network topology. (a) Mean delay time. (b) Mean number of stops. (c) Mean normalised delay time. (d) Mean normalised number of stops. (e) Maximum delay time.*

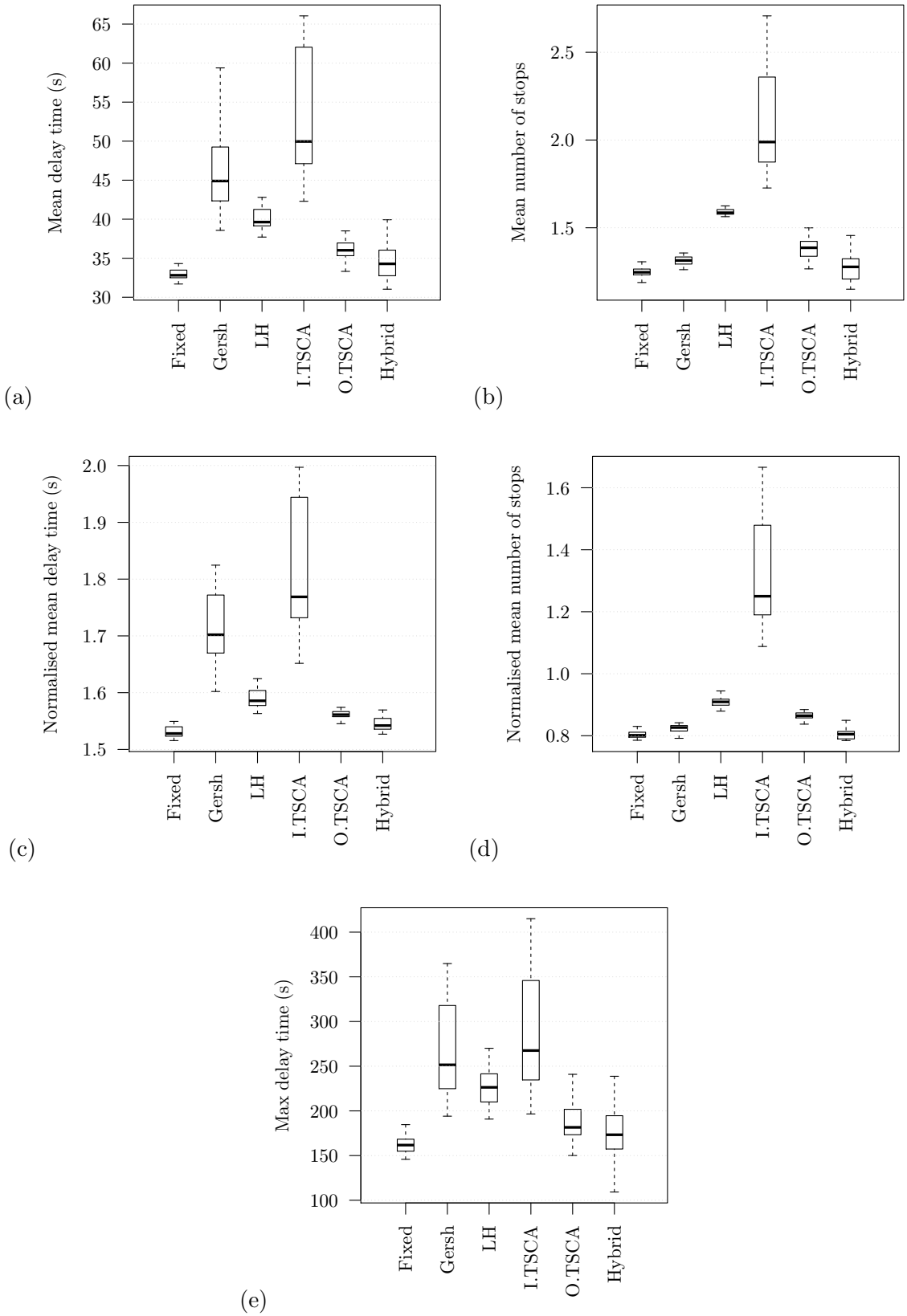


FIGURE 7.7: *Results for $\lambda = 30$ vehicles per minute for the corridor road network topology. (a) Mean delay time. (b) Mean number of stops. (c) Mean normalised delay time. (d) Mean normalised number of stops. (e) Maximum delay time.*

Simulation results for time-dependant vehicle arrival rates

This section contains simulation results for the case where the average vehicle arrival rate fluctuates as a function of time. The results are presented in Figures 7.8, 7.9 and 7.10, comparing the six traffic control algorithms in terms of mean roadway saturation levels, mean delay time and mean number of stops, respectively, for the road traffic corridor. Recording of the various performance measure indicators occurred every ten seconds during each simulation run. In Figure 7.8 it may be observed that the fluctuations in mean roadway saturation levels mimic those of the average arrival rate, as expected. Under the initial light traffic flow conditions (*i.e.* from 1 800 seconds to 3 000 seconds), the performances of the six traffic signal control algorithms are relatively similar, as indicated by their close grouping, although Hybrid may be observed to exhibit the lowest mean roadway saturation value. This is largely due to the ability of Hybrid to utilise the gaps between vehicles (which are relatively large, and numerous for light traffic flow) to maximise intersection utilisations.

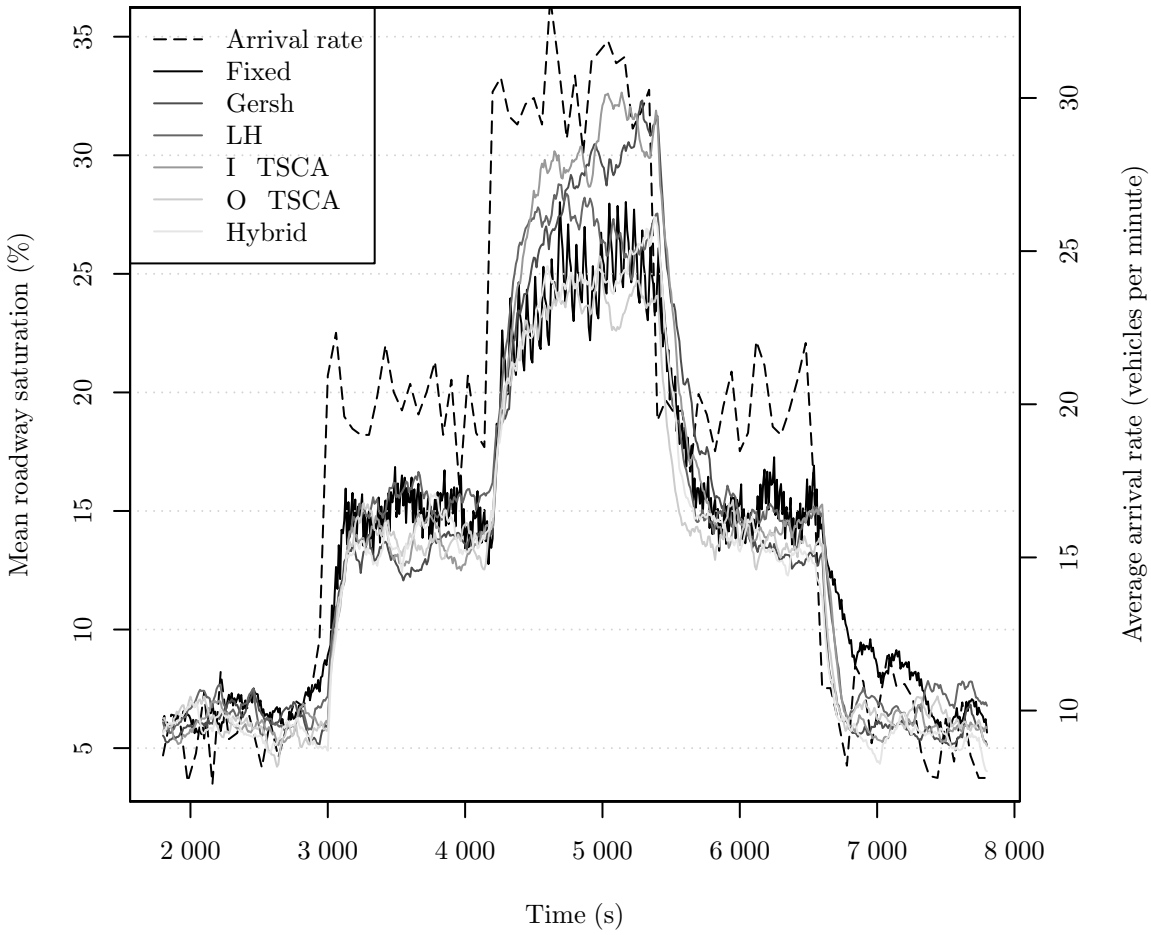


FIGURE 7.8: *The mean corridor road way saturation in relation to the average vehicle arrival rate.*

During the following period of medium traffic flow (*i.e.* from 3 000 seconds to 4 200 seconds), there is more variation in the performances of the algorithms, with Gersh, I-TSCA, O-TSCA and Hybrid outperforming Fixed and LH. This is again due to the improved flexibility of the former four algorithms over the latter two.

During the period of high traffic flow (*i.e.* from 4 200 seconds to 5 400 seconds), O-TSCA may

be observed to be the best performing algorithm. The inflexible green times of Fixed result in large fluctuations in its performance. The performance of Gersh declines significantly during this period due to the implementation of excessively long green times, which result in vehicle-queue spill-backs and considerably larger roadway saturation values.

As the average arrival rate falls to one consistent with medium traffic flow over the next twenty-minute period (*i.e.* from 5 400 seconds to 6 600 seconds), the performances of the algorithms become more similar once again. Fixed may be observed to be the worst performing algorithm. This is due to the fact that the signal timing settings implemented during the period are calculated assuming medium traffic flow and are therefore not able to effectively clear the relatively large residual number of vehicles which remain present on the roadway following the period of high traffic flow. This stands in stark contrast to the performance of O-TSCA, which results in the sharpest drop in roadway saturation. This is due to the fact that O-TSCA continues to implement relatively long green times even after the drop in the average arrival rate. Approximately halfway through the period, however, Gersh and Hybrid appear to outperform O-TSCA. This occurs once all residual queues have been cleared following the period of high traffic flow, at which point in time a more flexible traffic control approach becomes more appropriate.

A similar pattern prevails in the last period (*i.e.* from 6 600 seconds to 7 800 seconds) to that in the previous period. Hybrid and O-TSCA both relieve saturation in the most timely manner following the drop in average arrival rate, but shortly after this initial drop, Hybrid outperforms O-TSCA as a result of improved flexibility, facilitated by the IUMSM. Once again, it may be observed that Hybrid is the best performing algorithm under light traffic flow conditions in terms of minimising roadway saturation.

In Figure 7.9, there is a clear distinction between the performances of the six different traffic signal control algorithms with respect to the mean delay time experienced by vehicles. It may be observed that the performance of Hybrid is considerably superior to those of the other five algorithms for the initial twenty-minute period (*i.e.* from 1 800 seconds to 3 000 seconds) of light traffic flow following the warm-up period. The performances of Gersh and I-TSCA are very similar for the same period, as are the performances of LH and O-TSCA, while Fixed is the worst performing algorithm for the period. These results clearly illustrate the superior ability of the more flexible traffic signal control algorithms at reducing vehicle delay time under light traffic flow conditions.

During the following twenty-minute period (*i.e.* from 3 000 seconds to 4 200 seconds) of medium traffic flow conditions, the mean delay time value associated with each algorithm increases steadily, with LH and I-TSCA exhibiting the most pronounced rates of increase. The performances of Hybrid and Gersh become very similar towards the end of the period, illustrating that Gersh exhibits improved adaptation to the increased average vehicle arrival rate.

During the period of heavy traffic flow (*i.e.* from 4 200 seconds to 5 400 seconds), the rate of increase in the mean delay time of each algorithm becomes even more pronounced, most notably for Gersh, LH and I-TSCA. In fact, by the end of the period, these three algorithms are all outperformed by Fixed and O-TSCA. The poor performances of LH and I-TSCA may be attributed to the implementation of green times which are too short for the prevailing traffic flow conditions, resulting in frequent signal switches and thus, considerable intersection underutilisation. On the other hand, the poor performance of Gersh may be attributed to assigning green times which are too long, resulting in queue spill-backs. The improvement in the performance of Fixed relative to the three aforementioned algorithms is due to the increased uniformity of arriving vehicles, which lends itself towards fixed traffic signal control. Hybrid and O-TSCA are able to achieve a balance of suitably long green time periods which are implemented at appropriate times so as to maximise intersection utilisation. Hybrid is able to continue to outperform O-TSCA as its

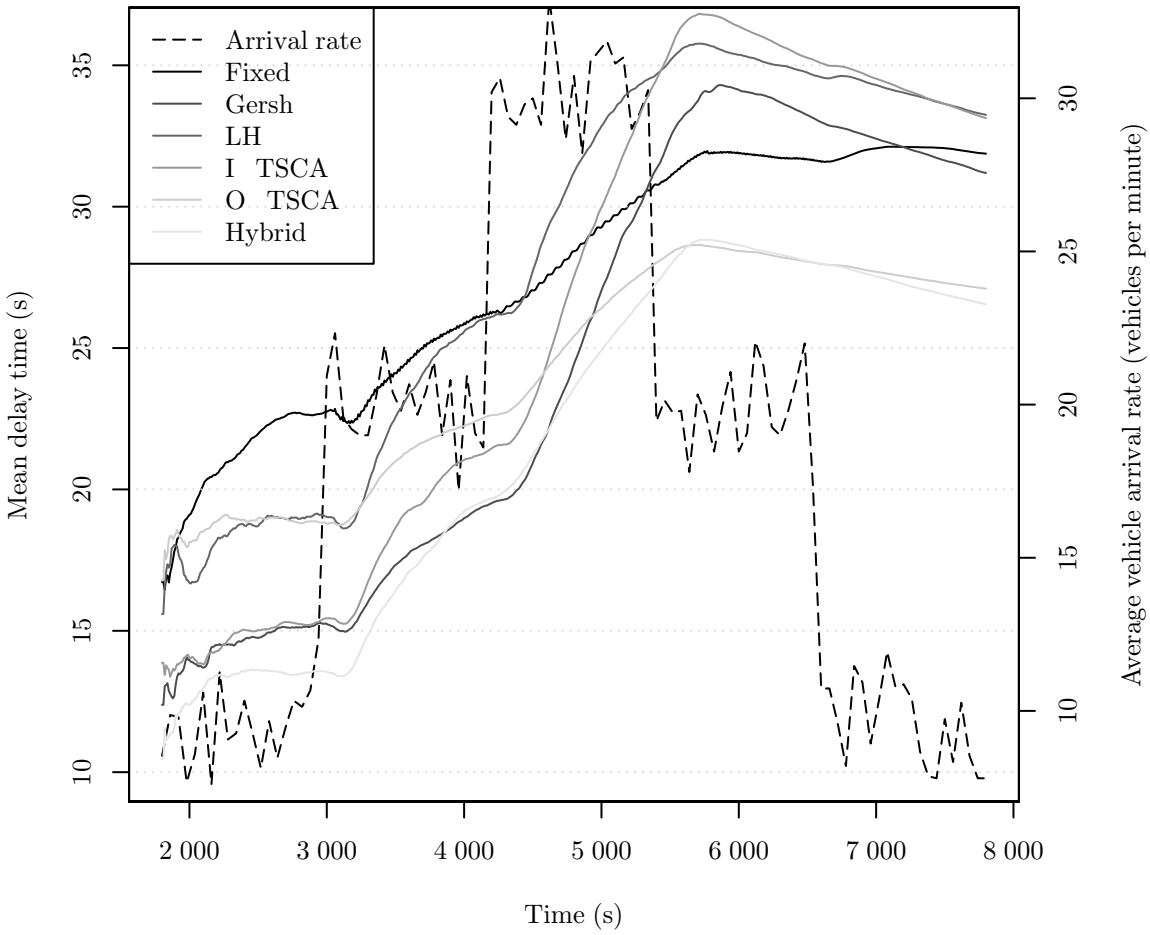


FIGURE 7.9: *The mean delay time experienced in relation to the average vehicle arrival rate to the corridor.*

IUMSM enables it to take advantage of any gaps in traffic arriving at the intersections so as to further increase intersection utilisation and reduce delay time.

The mean delay time values associated with each algorithm continue to rise briefly into the following twenty-minute period (*i.e.* from 5 400 seconds to 6 600 seconds) in spite of a decrease in the average vehicle arrival rate. This is due to the presence of relatively large numbers of vehicles still present along the roadways following the period of high traffic flow conditions. O-TSCA appears to momentarily outperform Hybrid. This is due to the fact that O-TSCA provides longer green times, allowing for the efficient clearing of vehicles while the IUMSM of Hybrid may terminate green times prematurely. Towards the end of the period, however, their respective performances appear to be identical. This may be attributed to the number of vehicles along the roadways decreasing, resulting in larger gaps between vehicles, of which the IUMSM of Hybrid may take advantage to increase intersection utilisation, compared to O-TSCA. The rate of improvement in mean delay time is least pronounced for Fixed. This is because Fixed is the least flexible of the algorithms and is therefore not able to adjust sufficiently to the residual vehicles remaining following the drop in average arrival rate.

A resumption in light traffic flow conditions during the final twenty-minute period (*i.e.* from 6 600 seconds to 7 800 seconds) results in the performances of the more flexible traffic signal control algorithms returning to the fore. Hybrid is distinguished as the superior control algorithm over

O-TSCA by the end of the period. LH, Gersh and I-TSCA all exhibit marked improvements in reducing mean delay time over Fixed in particular, and O-TSCA to a lesser extent, but due to their poor performances under heavy traffic flow conditions they are all considerably outperformed by O-TSCA by the end of the analysis period. From these results, it may be observed that Hybrid is superior in terms of minimising vehicle delay time under varying traffic flow conditions.

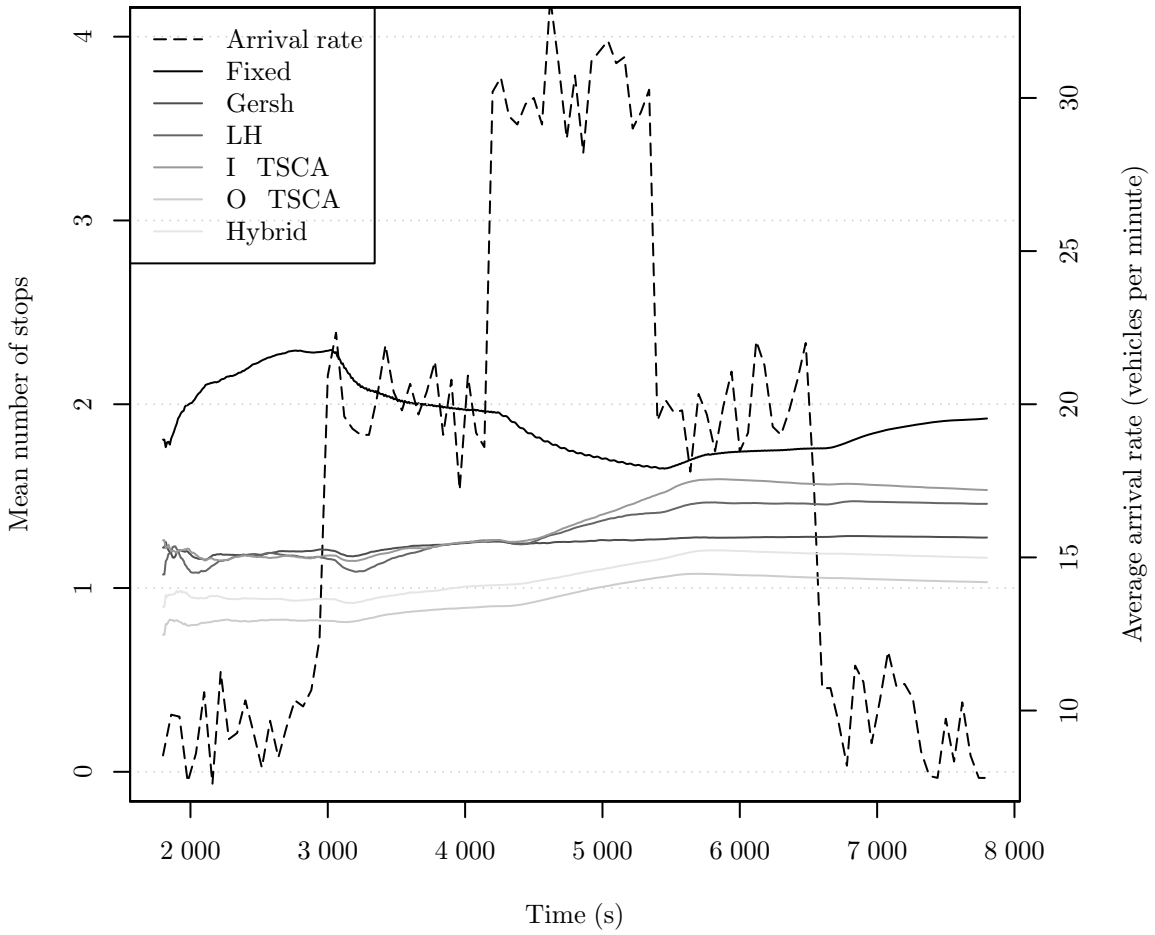


FIGURE 7.10: *The mean number of stops made in relation to the average vehicle arrival rate to the corridor.*

Figure 7.10 depicts the results pertaining to the number of stops associated with the implementation of the six traffic control algorithms compared in this study. As with the results pertaining to the mean delay time, there is a clear distinction between the performances of the different algorithms. During the first twenty-minute period (*i.e.* from 1 800 seconds to 3 000 seconds), when light traffic flow conditions prevail, it may be observed that O-TSCA is the best performing algorithm, followed by Hybrid, both of which exhibit mean numbers of stops smaller than one. The performances of Gersh, LH and I-TSCA are all relatively similar, resulting in vehicles requiring to stop marginally more than once, on average. The worst performing algorithm is Fixed. From these results it may be seen that the longer the green time duration implemented by a traffic control algorithm, the more improved the level of coordination between intersections it achieves, as O-TSCA implements the longest green time durations, on average, while Fixed implements the shortest.

During the the next twenty-minute period (*i.e.* from 3 000 seconds to 4 200 seconds), when

medium traffic flow conditions prevail, Fixed is the only algorithm which exhibits an improvement in terms of reducing its associated mean number of stops value. The reason for this is that Fixed implements longer green times which are better suited to the increase in average vehicle arrival rate. The associated mean numbers of stops of the other five algorithms increase marginally throughout the period. Noticeably, the performances of LH, I-TSCA and Gersh converge towards the end of the period as a result of the green times implemented by Gersh becoming longer in duration while those of LH and I-TSCA remain insufficiently short. During the period of heavy traffic flow (*i.e.* 4 200 seconds to 5 400 seconds), the improvement in the performance of Fixed increases as a result of a further increase in the green time durations implemented while the performances of LH and I-TSCA continue to decline.

For the penultimate twenty-minute period (*i.e.* from 5 400 seconds to 6 600 seconds), all the algorithms exhibit marginal improvements in terms of reducing the mean number of stops made by vehicles, save for Fixed. This is because Fixed implements shorter green times upon commencement of the medium traffic flow period, before the relatively large number of vehicles present on the road network (as a result of the previous period of heavy traffic flow) has decreased. This pattern continues throughout the final period (*i.e.* from 6 600 seconds to 7 800 seconds) of light traffic flow. From these results it may be observed that O-TSCA is superior in terms of facilitating coordination between intersections along the corridor, thereby reducing the mean number of stops made by vehicles.

7.2.3 A 3×4 grid of 12 homogeneous intersections

In this section, results and analyses are presented pertaining to a 3×4 grid road network topology, first for fixed constant vehicle arrival rates and then for varying, time-dependent vehicle arrival rates. Figure 7.11 depicts the 3×4 grid of homogeneous intersections implemented within the microscopic traffic simulation modelling framework described in §4.2.

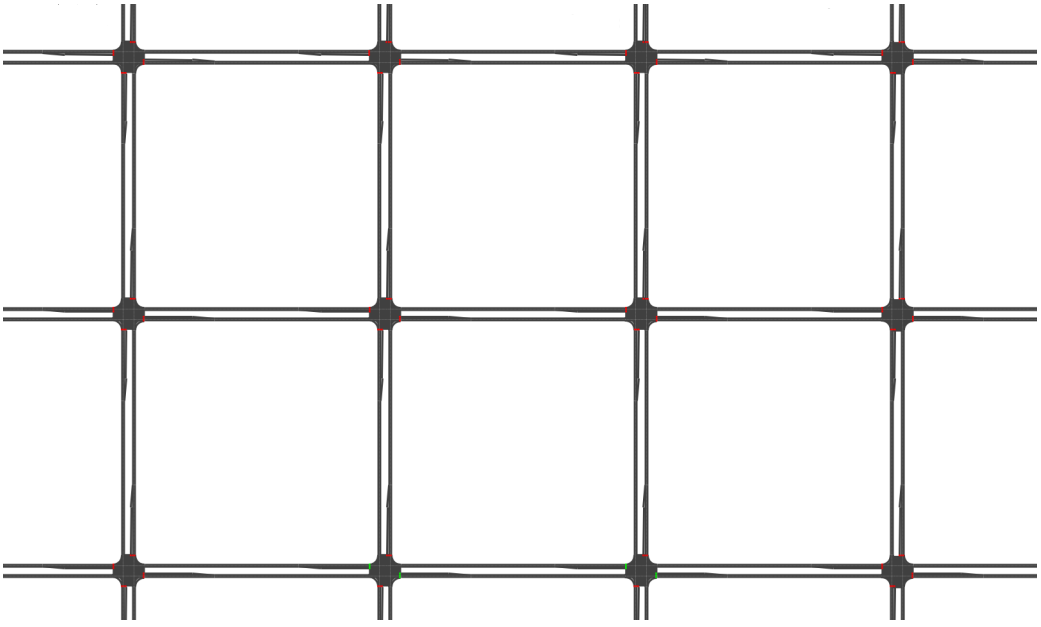


FIGURE 7.11: *The 3×4 grid of road traffic intersections implemented in §7.2.3 within the microscopic traffic simulation modelling framework described in §4.2.*

Simulation results for light traffic flow conditions ($\lambda = 10$ vehicles per minute)

For the performance measures of mean delay time and mean normalised delay time, Hybrid was found to be the best performing algorithm (with an average mean delay time of 26.3 seconds and a mean normalised delay time of 1.26), as may be seen in Figures 7.12(a) and 7.12(c). At a 95% confidence level, Hybrid significantly outperformed Gersh, LH and Fixed by 3.6%, 28.8% and 20.9%, respectively. With respect to the mean normalised delay time, Hybrid may be expected to outperform Gersh (the next best performing algorithm) by 0.7% at a 95% confidence level. Gersh, however, did achieve the lowest maximum delay time of 114.6 seconds as well as the lowest average maximum delay time of 95.3 seconds (see Figure 7.12(e)), although this was not a statistically significant improvement over that of Hybrid at a 95% confidence level.

When considering the mean number of stops made by vehicles, it was found that the performance of O-TSCA was superior, as shown in Figure 7.12(b), outperforming Hybrid, Gersh and Fixed by 35.9%, 47.6% and 64.1%, respectively. The same patterns of performance prevail in terms of the mean normalised number of stops (see Figure 7.12(d)), with O-TSCA exhibiting a value of 0.19, which translates to vehicles only requiring to stop, on average, at one out of every five intersections they encounter, approximately. It is also worth noting that the performances of Hybrid, with respect to the mean number of stops and mean normalised number of stops are statistically superior, at a 95% confidence level, to those of Gersh by margins of 18.3% and 24.2%, respectively.

LH is the worst performing algorithm in terms of mean delay time and normalised delay time, as well as maximum delay time, as may be seen in Figures 7.12(a), 7.12(c) and 7.12(e), respectively. This is not due to the wrong allocation of green time (the average duration of the green times allocated by LH at the twelve intersections lay between those of Gersh and Hybrid), but the wrong time of implementation for these green times, *i.e.* LH is less effective than Gersh and Hybrid at utilizing gaps in vehicle platoons. While LH appears to cause fewer vehicle stops and thus improved coordination when compared to Fixed, as indicated by Figures 7.12(b) and 7.12(d), the duration of these stops is comparatively longer, resulting in increased delay time. Gersh and Hybrid are the most effective algorithms at reducing vehicle delay time as a result of their superior flexibility and ability to maximise intersection utilisation, but flexibility does not guarantee favourable results, as is evident from the performance of I-TSCA which is outperformed by O-TSCA in respect of all five performance measure indicators.

While Hybrid outperforms O-TSCA in terms of mean and maximum delay time as well normalised mean delay time, O-TSCA outperforms Hybrid in terms of mean and mean normalised number of stops made. Based on the margins of these differences in performance, it would appear that O-TSCA is the superior traffic control algorithm under light traffic flow conditions for the 3×4 grid of road traffic intersections.

Simulation results for medium traffic flow conditions ($\lambda = 20$ vehicles per minute)

Figure 7.13 indicates that O-TSCA statistically outperformed all five of the other algorithms for four of the five performance measure indicators at a 95% confidence level, and while it achieved a lower average maximum delay time than that of Hybrid (see Figure 7.13(e)), this difference was not considered statistically significant at a 95% level of confidence. In terms of mean delay time and normalised mean delay time, O-TSCA significantly outperformed Gersh by 15.1% and 3.9%, respectively (see Figures 7.13(a) and 7.13(c), respectively). With respect to the mean and mean normalised number of stops made, O-TSCA outperformed Hybrid by 34.2% (see Figure 7.13(b)) and 41.5% (see Figure 7.13(d)), respectively.

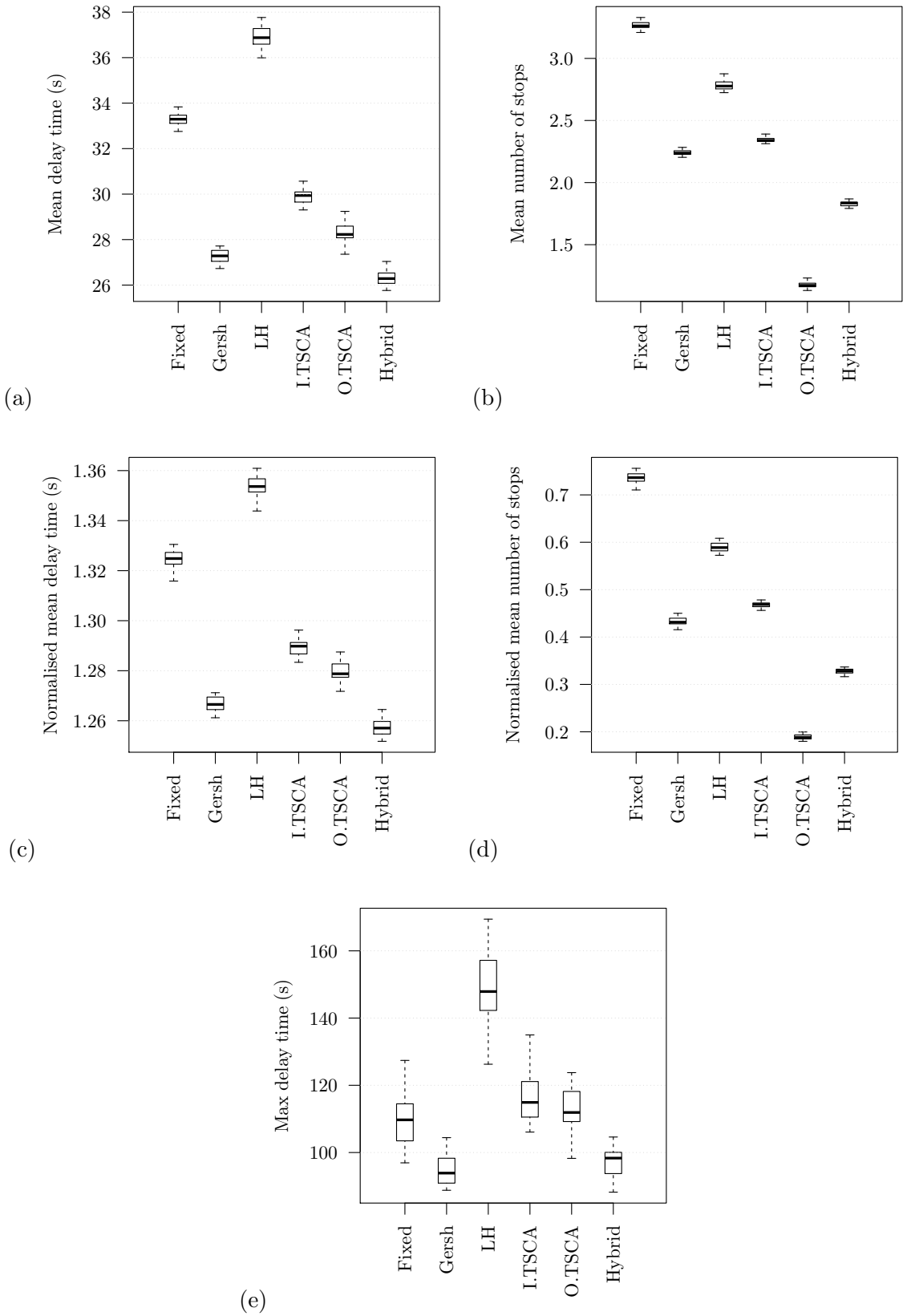


FIGURE 7.12: *Results for $\lambda = 10$ vehicles per minute for the grid road network topology. (a) Mean delay time. (b) Mean number of stops. (c) Mean normalised delay time. (d) Mean normalised number of stops. (e) Maximum delay time.*

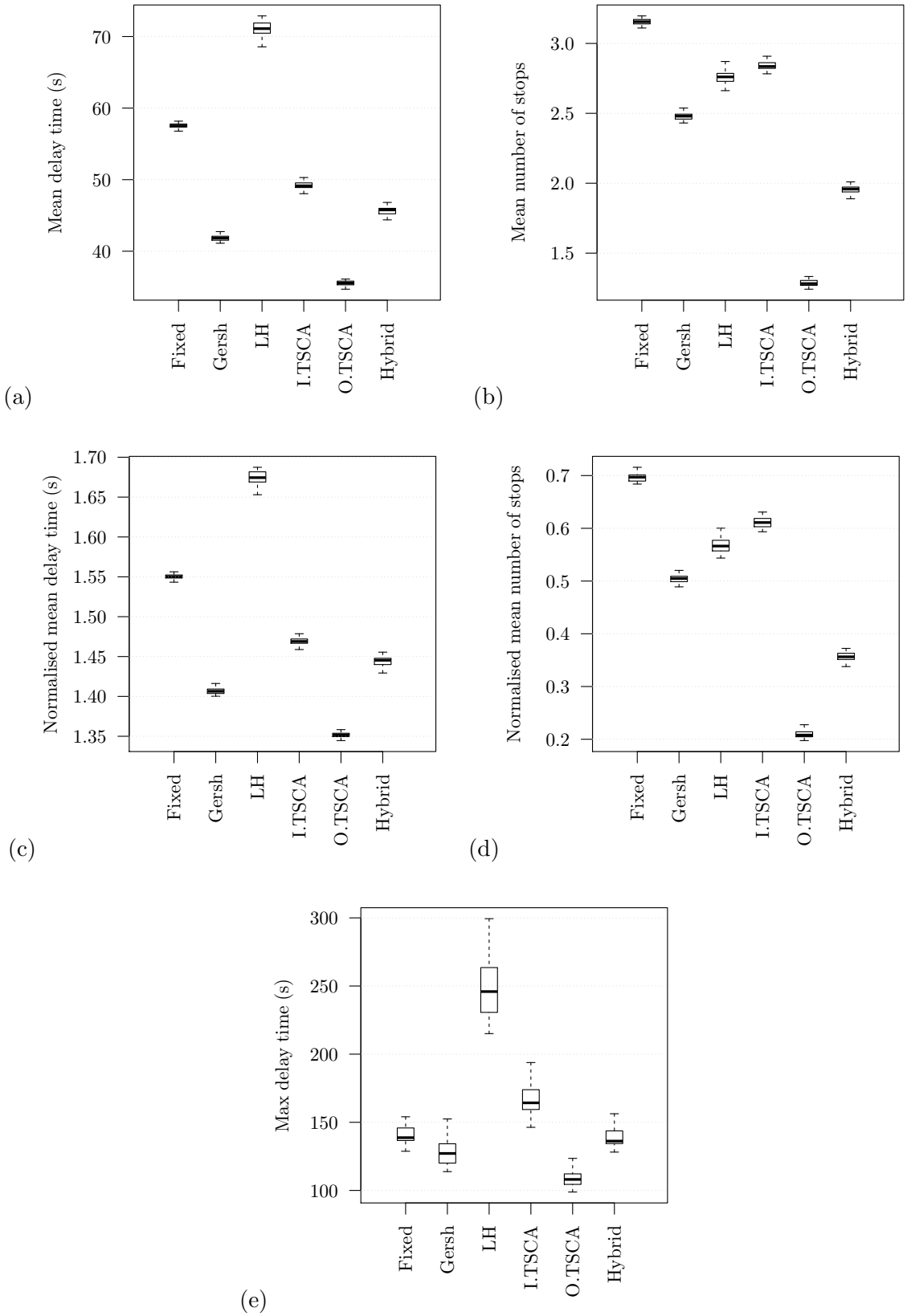


FIGURE 7.13: *Results for $\lambda = 20$ vehicles per minute for the grid road network topology. (a) Mean delay time. (b) Mean number of stops. (c) Mean normalised delay time. (d) Mean normalised number of stops. (e) Maximum delay time.*

These results obtained under medium traffic flow conditions are a direct result of the green times implemented by the various traffic control algorithms. Fixed implemented green times of 7 seconds at each of the 12 intersections. Gersh and I-TSCA implemented similar green times at intersections throughout the network, but they were also observed to implement green times for as long as 30 seconds. LH and Hybrid each implemented green times of approximately 15 seconds on average. It was observed, however, that in some instances LH would implement green times which lasted less than a second, resulting in significant intersection underutilisation and increased delay times. O-TSCA, on the other hand, implemented average green times of approximately 25 seconds at each intersection, with minimum average green times of approximately 15 seconds.

These results illustrate that merely implementing appropriate green times is not sufficient for reducing vehicle delay time and facilitating coordination between intersections, as is indicated by the differences in performance between Fixed, Gersh and I-TSCA, as well as between Hybrid and LH. Furthermore, it may be observed that control algorithms which rely on the prediction of values for input data (*i.e.* LH and I-TSCA) appear to be inferior to those which rely on on live actual vehicle data, such as Gersh and O-TSCA. As with light traffic flow conditions, O-TSCA is considered to be the superior traffic control algorithm for medium traffic flow conditions for the 3×4 grid of road traffic intersections.

Simulation results for heavy traffic flow conditions ($\lambda = 30$ vehicles per minute)

As may be seen from Figure 7.14(a), O-TSCA exhibited a mean delay time value of 52.18 seconds, which is statistically superior to those of the other five algorithms, at a 95% confidence level. In particular, O-TSCA outperformed Fixed (the second best performing algorithm) by 17.3%. The results pertaining to the mean normalised delay time, shown in Figure 7.14(c), are much the same. Although Fixed achieved the lowest average maximum delay value of 155.77 seconds, as illustrated in Figure 7.14(e), this was not considered to be statistically superior to that of O-TSCA at a 95% confidence level.

O-TSCA proved to be the best performing algorithm with respect to the mean and mean normalised number of stops, as illustrated in Figures 7.14(b) and 7.14(d), respectively. It achieved a mean number of stops value of 1.91 which is a 12.6% and 17.9% improvement over Hybrid and Fixed, respectively, at a 95% level of confidence. In terms of the mean normalised number of stops, on the other hand, O-TSCA achieved a value of 0.34, which was again statistically superior to Hybrid and Fixed, at a 95% confidence level, by margins of 17.7% and 22.5%, respectively.

As a result of traffic flow increasing and becoming more constant, the traffic control algorithms which implemented longer green times and therefore fewer signal switches (*i.e.* Fixed, O-TSCA and Hybrid) exhibited improved results over the more flexible traffic control algorithms (*i.e.* LH and I-TSCA). Gersh was outperformed for all five performance measure indicators by Fixed and Hybrid despite implementing green times that were very similar, on average, to those of Fixed and Hybrid. The reason for this was that on several occasions, these longer green times implemented by Gersh eventually led to grid-lock. Excess green time was awarded to intersection approaches before the regulatory measures of the algorithm could terminate it. This led to queue spill-backs which blocked intersections. Furthermore, the logic rules of the algorithm prevented the grid-lock situation from being resolved once it occurred.

Based on the results of Figure 7.14, O-TSCA appears to be the superior algorithm under heavy traffic flow conditions for the 3×4 grid of road traffic intersections.

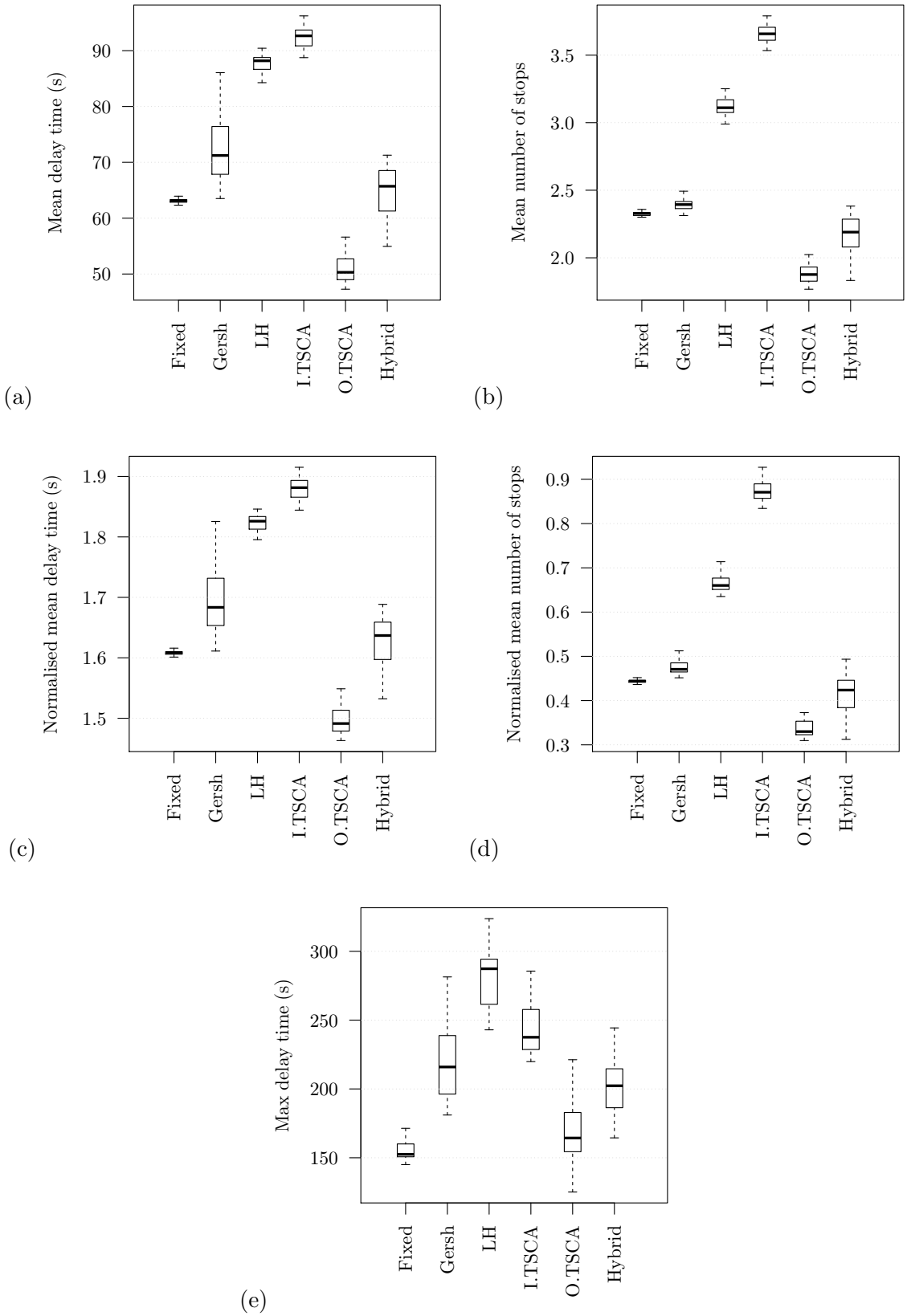


FIGURE 7.14: *Results for $\lambda = 30$ vehicles per minute for the grid road network topology. (a) Mean delay time. (b) Mean number of stops. (c) Mean normalised delay time. (d) Mean normalised number of stops. (e) Maximum delay time.*

Simulation results for time-dependant vehicle arrival rates

This section contains simulation results for the case where the average vehicle arrival rate fluctuates as a function of time. The results are presented in Figures 7.15, 7.16 and 7.17, comparing the six traffic control algorithms in terms of mean roadway saturation levels, mean delay time and mean number of stops, respectively, for the the 3×4 grid road network topology. Recording of the various performance measure indicators occurred every ten seconds during each simulation run.

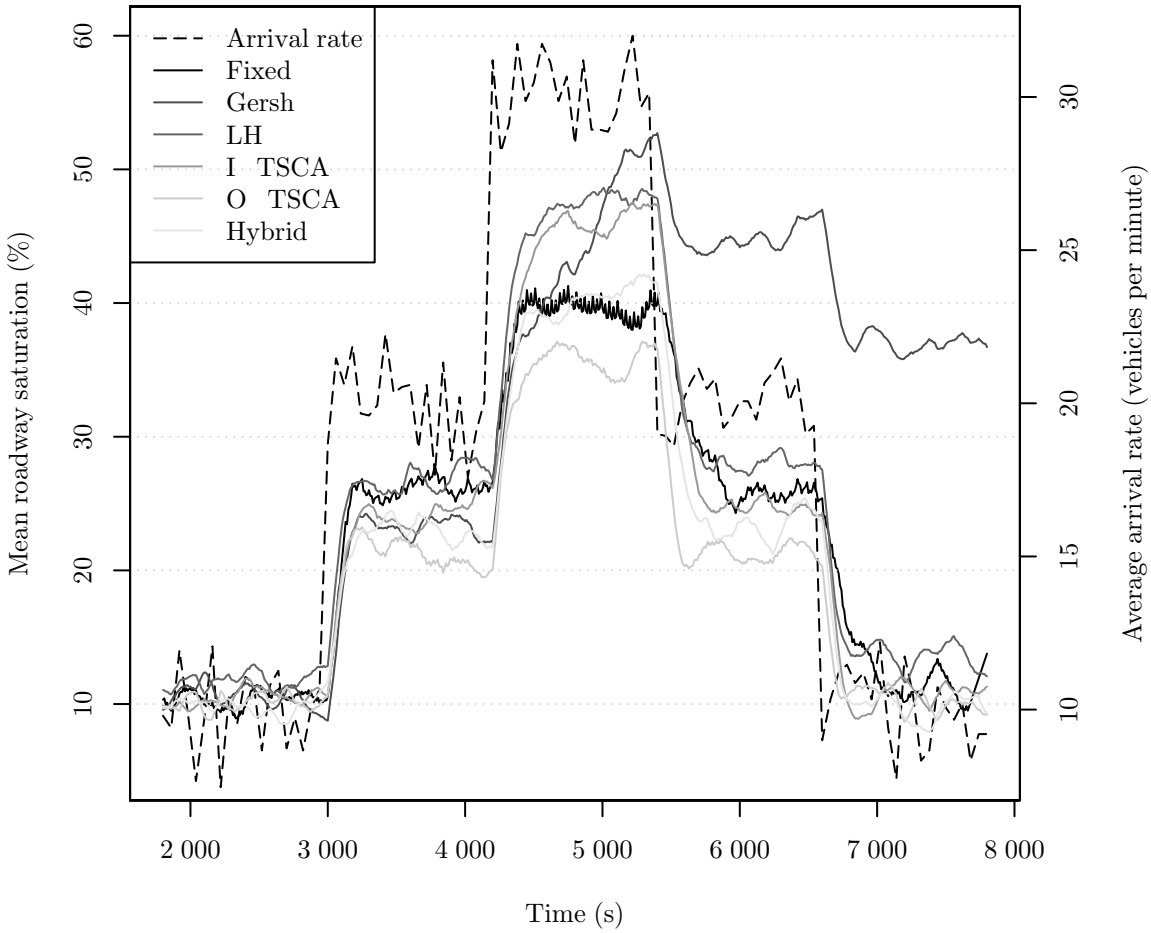


FIGURE 7.15: *The mean grid road way saturation in relation to the average vehicle arrival rate.*

In Figure 7.15 it may be observed that Hybrid is the best performing algorithm for the initial twenty-minute period (*i.e.* from 1 800 seconds to 3 000 seconds) of light traffic flow following the warm-up period. Gersh, however, maintains an average roadway saturation value of less than 10% for the longest period of time, before it increases as a result of an increase in the average vehicle arrival rate. The performances of the remaining four algorithms are comparatively similar, with LH appearing to be the worst performing algorithm.

During the twenty-minute period of medium traffic flow (*i.e.* from 3 000 seconds to 4 200 seconds), O-TSCA appears to outperform the other five algorithms by a large margin. This is a result of the improved coordination between intersections achieved by O-TSCA which enables vehicles to reach their destinations within a shorter period of time, thereby reducing the saturation levels along the roadways. The performances of the three most flexible algorithms, namely Gersh, Hybrid and I-TSCA, are comparatively similar for the majority of the period, before the

performance of I-TSCA declines in relation to those of Gersh and Hybrid. This is due to I-TSCA switching too frequently between phases, while Gersh and Hybrid are able to implement longer green times.

For the twenty-minute period of heavy traffic flow (*i.e.* from 4 200 seconds to 5 400 seconds), the improvement in the performance of O-TSCA over the other five algorithms becomes considerably more pronounced. By the end of the period, the second best performing algorithm is, in fact, Fixed, which even outperforms Hybrid. This is due to the increased green time durations implemented by Fixed in comparison to Hybrid and the regular frequency at which they occur. Gersh exhibits the most noticeable decline in performance during the period, resulting in it becoming the poorest performing algorithm overall. This degeneration in performance is attributed to the fact that in several instances, the implementation of Gersh led to network-wide gridlock. This was due to the fact that the algorithmic logic of Gersh allows for queue spill-backs to block upstream intersections, thus preventing vehicles from entering them. Furthermore, it was observed that once these gridlocks occurred, the algorithm was unable to resolve them, resulting in considerably large saturation levels.

During the fourth twenty-minute period (*i.e.* from 5 400 seconds to 6 600 seconds) the mean roadway saturation value associated with O-TSCA drops rapidly by approximately 15%, on average, following a decrease in the average vehicle arrival rate. This illustrates that not only is O-TSCA the most effective algorithm at prolonging an increase in the mean roadway saturation levels following an increase from medium traffic flow to high traffic flow, but also the most effective algorithm at alleviating mean roadway saturation levels following a decrease in average vehicle arrival rates. Hybrid and I-TSCA may also be seen to be associated with significant drops in mean roadway saturation, eventually outperforming Fixed once again by the end of the period. While Gersh does result in a decrease in mean roadway saturation during the period, it is still convincingly outperformed by the other five algorithms as a result of gridlock occurring sporadically.

During the final twenty-minute period (*i.e.* from 6 600 seconds to 7 800 seconds) of light traffic flow conditions, Hybrid once again outperforms the more rigid O-TSCA. This may be attributed to its improved intersection utilisation, as a result of switching between signals during suitably large gaps between arriving vehicles. I-TSCA may also be observed to outperform Fixed and LH as a result of its superior flexibility. While it may appear that Gersh is the worst performing algorithm by a considerable margin, it should be noted that in the instances when Gersh was implemented and gridlock did not occur, its performances were very similar to those of Hybrid.

In Figure 7.16, the differences in performances of the six algorithms in respect of mean delay time are clearly distinguishable. During the first twenty-minute period (*i.e.* from 1 800 seconds to 3 000 seconds), when light traffic flow conditions prevail, Hybrid and Gersh are the two best performing algorithms while LH is the worst. This is largely attributed to the poor intersection utilisation of LH.

It may be observed during the second twenty-minute period (*i.e.* from 3 000 seconds to 4 200 seconds) that the mean delay time associated with O-TSCA increases initially following an increase in the average vehicle arrival rate, before remaining relatively constant during the remainder of the period. This stands in contrast to the other five algorithms whose associated mean delay time values increase continuously throughout the period. This is again attributed to the improved coordination between intersections achieved by O-TSCA and fewer signal switches as a result of longer green times. As a result, O-TSCA may be seen to start outperforming Hybrid and Gersh about halfway through the period. At the same point in time, Gersh appears to begin outperforming Hybrid.

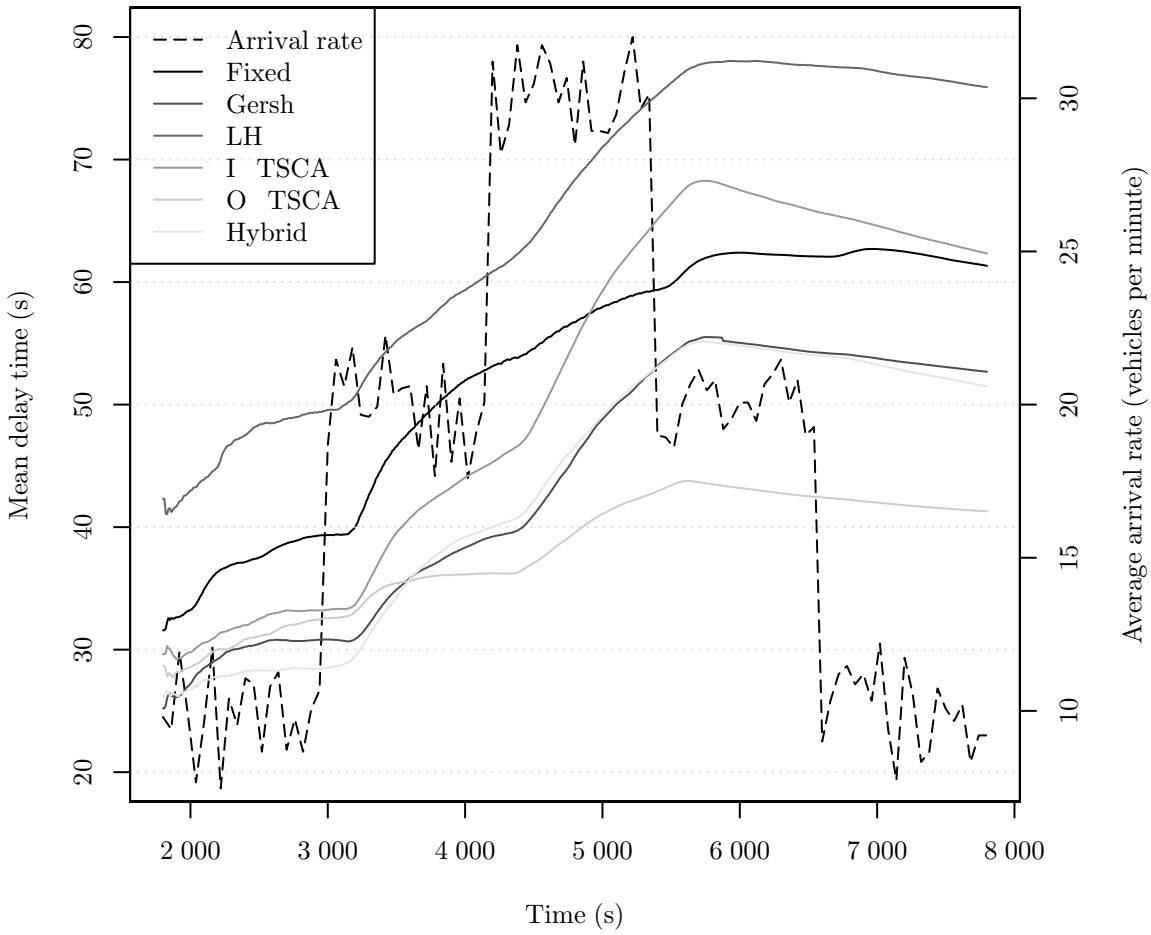


FIGURE 7.16: *The mean delay time experienced in relation to the average vehicle arrival rate to the grid.*

The associated mean delay times of all six algorithms continue to rise during the twenty-minute period of heavy traffic flow (*i.e.* from 4 200 seconds to 5 400 seconds) with O-TSCA and Fixed exhibiting the least steep increase due to comparatively longer green time durations.

Over the final forty-minute period of medium traffic flow and light traffic flow conditions, the associated delay times of all six algorithms decline, with I-TSCA exhibiting the steepest rate of decline, while both Fixed and LH exhibit relatively gentle rates of decline. By the end of the final twenty-minute period it may be observed that Hybrid marginally outperforms Gersh. From the results of Figure 7.16, it may be seen that in spite of being outperformed by both Gersh and Hybrid during the initial twenty-minute period of light traffic flow conditions, O-TSCA is the superior traffic signal control algorithm in terms of reducing mean delay time for the grid network topology while the vehicle arrival rate fluctuates over time.

In Figure 7.17, it may be observed that the performance of O-TSCA is superior with respect to minimising the mean number of stops made to the other five algorithms for the entire analysis period (*i.e.* from 1 800 seconds to 7 800 seconds). This is an indication that for the grid network topology, O-TSCA is the best performing algorithm at facilitating coordination between intersections, regardless of the vehicle arrival rate and traffic flow conditions. The mean number of stops associated with O-TSCA, Gersh and Hybrid remain relatively constant throughout the entire analysis period. This result, together with the results of Figures 7.15 and 7.16, indicate that the three algorithms are effective at adjusting their green time durations to adapt to the

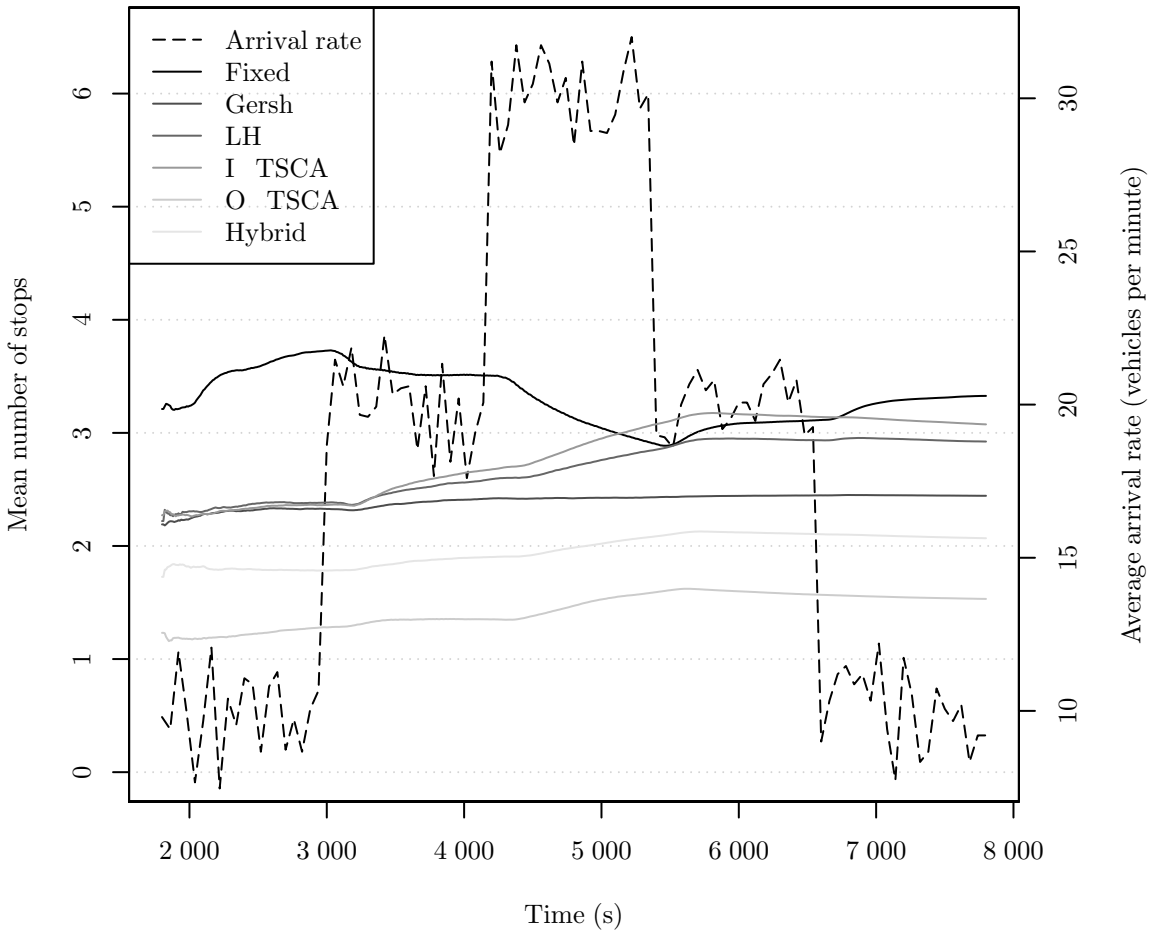


FIGURE 7.17: *Mean number of stops made for varying average arrival rates to the grid.*

prevailing traffic conditions. In other words, an increase or decrease in vehicle arrival rates does not result in vehicles requiring to stop more or less frequently, but rather the duration of the stops varies.

On the other hand, from the second twenty-minute period (*i.e.* from 3 000 seconds to 4 200 seconds) onwards, the mean number of stops associated with LH and I-TSCA increases and continues to do so throughout the period of high traffic flow (*i.e.* from 4 200 seconds to 5 400 seconds). This result, together with an increase in mean roadway saturation and mean delay time (as illustrated in Figures 7.15 and 7.16, respectively), indicate that the algorithms are not effective at adjusting to increases in vehicle arrival rates and traffic flow conditions.

Finally, it may be observed that the associated mean number of stops of Fixed decreases during the second twenty-minute period (*i.e.* from 3 000 seconds to 4 200 seconds) and during the third twenty-minute (*i.e.* from 4 200 seconds to 5 400 seconds), before increasing again during the final forty minutes of the analysis period (*i.e.* from 5 400 seconds to 7 800 seconds). These fluctuations correspond to increases and decreases in the green time durations implemented by Fixed, respectively. As the green time durations decrease, Fixed is able to facilitate coordination between intersections, resulting in vehicles being required to make more stops, more often.

7.3 Traffic signal control algorithm appraisals

This section contains an appraisal of the performances of the six different traffic signal control algorithms considered in this chapter (in terms of the various performance measure indicators considered) for both the corridor and grid road network topologies, as well as for fixed and varying vehicle arrival rates.

Fixed. For both the corridor and grid network topologies, Fixed, in general, did not perform as well under light traffic flow conditions as the other five, more flexible traffic control algorithms. The reason for this may be attributed to the fact that under light traffic flow conditions, the green time durations implemented by Fixed are generally shorter than those of the better performing algorithms. This makes it considerably challenging for Fixed to facilitate any kind of coordination between intersections, resulting in vehicles having to make a larger number of stops and incurring greater delay times, as is illustrated by the results in §7.2.1 and §7.2.2. For medium and high traffic flow conditions, on the other hand, Fixed performs significantly better. In fact, for the case of the the corridor network topology, the performance of Fixed was found to be indistinguishable from that of Hybrid, the best performing self-organising traffic control algorithm, at a 95% confidence level. The reason for this improvement is that under heavier traffic flow conditions, the vehicle arrivals at an intersection become more constant and uniform, which suits fixed traffic signal control as they more closely replicate the assumed average arrival rates which are used to determine the green time durations of the algorithm.

Gersh. Gersh performed relatively well under light and medium traffic flow conditions for both road network topologies, particularly in terms of reducing mean vehicle delay time. Gersh is a very flexible traffic control algorithm which, under lighter traffic flow conditions, results in relatively short green time durations and frequent switching of traffic signals. Vehicles may therefore be required to stop more often, but these stops will be relatively short in duration. The disadvantage of Gersh is that under higher traffic flow conditions, it implements excessively long green time durations. These excessively long green time durations often result in queue spill-backs which block upstream intersections, thus preventing other vehicles from entering them. For the grid network topology, these spill-backs often lead to gridlock. While there is a mechanism in the algorithmic logic of Gersh for preventing this from happening (as was described in §5.3.1), it is not effective under the more realistic testing conditions of the microscopic traffic simulation modelling framework of this study. For a green time to be terminated, this mechanism requires a vehicle to be completely stopped a short distance from the intersection on an adjoining departure lane. It was observed, however, that in the framework implemented in this study, by the time the vehicle had come to a complete stop, several vehicles had already entered the intersection behind it, and were then forced to wait in the intersection once the traffic signal had turned red.

LH. From the results of §7.2.1 and §7.2.2, it may be concluded that LH is better suited to the corridor network topology, than the grid network topology. Like Gersh, LH appears to perform better under lighter traffic flow conditions as opposed to heavier traffic flow conditions. It was found to be comparatively less flexible than Gersh and Hybrid. This may be attributed to both its stabilisation strategy (which is based upon a fixed-time control algorithm), and the fact that its optimisation strategy favours the traffic approach currently receiving service. Under light traffic flow conditions, these two factors result in LH implementing comparatively longer green time durations, resulting in improved coordination between intersections, but at the cost of increased vehicle delay times.

I-TSCA. From the results presented in §7.2.1 and §7.2.2, it may be seen that I-TSCA outperforms LH in all but one scenario (*i.e.* under heavy traffic flow for the grid network topology). This may be attributed to the fact that I-TSCA is more flexible than LH and therefore adapts better to various prevailing traffic flow conditions. While it was able to outperform LH, I-TSCA was unable to outperform Gersh under lighter traffic flow scenarios. Similarly, it is conceded that I-TSCA is not as effective as Fixed at reducing vehicle time or the number of stops made by vehicles under heavier traffic flow conditions. It is noted, however, that I-TSCA is not consistently outperformed by any of the three existing traffic signal control algorithms tested in this study for each of the scenarios investigated. In general, it may be concluded that I-TSCA is more effective at reducing vehicle delay time and number of stops made under lighter prevailing traffic signal conditions. A reason for this is that under heavier prevailing traffic signal conditions, I-TSCA tends to terminate green times prematurely, leading to a lack of coordination, and therefore increased vehicle delay time and number of stops made. This phenomenon is more prevalent in the results presented in §7.2.2 for the grid network topology.

O-TSCA. In comparison to Gersh, I-TSCA and Hybrid, O-TSCA is a less flexible traffic signal control algorithm. This lack of flexibility is due to the implementation of relatively long green time durations. Under lighter prevailing traffic flow conditions, this lack of flexibility results in O-TSCA exhibiting relatively large vehicle delay times when compared to Gersh, Hybrid and I-TSCA. This is especially prevalent for the corridor network topology, as illustrated in §7.2.1 . These longer green times do, however, aid in the ability of O-TSCA to facilitate coordination between intersections. A further contributing factor towards the superior intersection coordination abilities of O-TSCA is that the algorithm does not only rely on vehicles approaching the intersection to inform its signal switching policies, but also on the space available for the vehicles to occupy once they have passed through the intersection. This prevents O-TSCA from providing excessively long green times which would lead to vehicle queue spill-backs. In other words, each intersection which implements O-TSCA essentially "packages" all approaching vehicles into platoons of manageable sizes for the next downstream intersection. The sizes of these platoons are determined by the amount of space available for the vehicles to occupy once they cross the intersection. Thus, an intersection will not pass on a platoon of vehicles which is too large for the following downstream intersection to provide service to without spill-backs occurring. It may be observed from the results of §7.2.1 and §7.2.2 that O-TSCA is considerably more effective at reducing both vehicle delay time and number of stops made, when implemented for the grid network topology. A reason for this is that an intersection which implements O-TSCA is more effective at serving platoons of vehicles, rather than individually arriving vehicles. Thus, the more intersections there are in a road network which can pack vehicles into manageable platoons, the more effective O-TSCA is. For the corridor road network topology, vehicles travelling along ten of the sixteen intersection approach lanes have not yet passed through another intersection, and therefore have not yet been packed into a platoon. For the grid road network topology, however, only the vehicles travelling along fourteen of the forty-eight intersection approaches would not have been packed into platoons yet. Based on the results presented in §7.2.2, it may be concluded that O-TSCA is the best overall performing traffic control algorithm for the grid network topology investigated in this study.

Hybrid. Of all six traffic signal control algorithms, Hybrid achieves the best balance between flexibility and coordination. It harnesses the flexibility of I-TSCA and its IUMSM to ensure that vehicles are served in as timely a manner as possible under lighter traffic flow conditions while it harnesses the coordination abilities of O-TSCA to ensure that

vehicles do not stop unnecessarily at intersections under heavier traffic flow conditions. The IUMSM ensures that green time durations are not too long under lighter traffic flow conditions and ensures that green times are not terminated prematurely under heavier traffic flow conditions. From the results presented in §7.2.1 and §7.2.2, it may be observed that Hybrid is more effective at reducing vehicle delay time and the number of stops made when implemented in a corridor network topology than in a grid network topology. That is not to say, however, that Hybrid is not effective when implemented for the grid network topology; it is just not as effective as O-TSCA. The reason for this is that between intersections, slight platoon dispersal tends to occur, in which case the IUMSM would terminate service to an approach before the originally detected platoon has passed through the intersection. This leads to the implementation of shorter green times, and therefore a decrease in coordination between intersections, when compared with O-TSCA. For the corridor network topology, however, when there is greater variation in the arrivals of vehicles at the intersections, it is concluded that it is more important for an algorithm to exhibit flexibility over coordination, and based on this, and the results of §7.2.1, it is concluded that Hybrid is the best overall performing traffic signal control algorithm for the corridor road network topology investigated in this study.

7.4 Chapter summary

This chapter opened with a description in §7.1 of the general parameters and settings applied during each simulation run. In particular, the approach followed to determine suitable warm-up periods was described in §7.1.1, while §7.1.2 was devoted to a discussion on the general model parameters and testing conditions. In §7.1.3, the values of the parameters specific to each traffic control algorithm were described for the various prevailing traffic scenarios, while the performance measures which were used to evaluate and compare the efficacies of the various traffic control algorithms were discussed in §7.1.4.

The results obtained were presented in §7.2. The Tukey Honest Significant Difference method was used to rank the algorithms and was described briefly in §7.2.1. The results obtained for a corridor comprising four homogeneous intersections were presented in §7.2.2, while the results obtained for a three-by-four grid of twelve homogeneous intersections were presented in §7.2.3. An appraisal of the relative performances of the traffic signal control algorithms investigated was finally presented in §7.3 .

CHAPTER 8

Conclusion

Contents

The dissertation closes with a summary of the work contained therein, an appraisal of the contributions of the dissertation and a discussion on possibilities for related future work.

8.1 Dissertation summary

The dissertation opened in Chapter 1 with a brief discussion on the debilitating effects of road traffic congestion and examples of strategies proposed for mitigating these effects. This discussion was followed by the description of the notions of self-organising systems and emergence systems, in partial fulfilment of Dissertation Objective 1(c) in §1.3. An informal description of the problem considered in this dissertation was also presented. The capabilities of recently developed radar detection sensors were discussed in terms of the vehicle data they are able to provide. These data were assumed as input for the various traffic signal control algorithms investigated in this dissertation. The chapter closed with a statement on the objectives to be pursued in the dissertation and a description of the organisation of material in the dissertation.

A literature review of existing traffic flow theory was conducted in Chapter 2, in fulfilment of Dissertation Objective 1(a) in §1.3. The chapter opened with an introduction to traffic flow theory in §2.1. This was followed by a discussion on microscopic traffic flow theory in §2.2, which included descriptions of various car-following models as well as the dynamics of vehicle delay at signalised intersections. The chapter closed in §2.3 in which macroscopic traffic flow theory. This section contained descriptions of the fundamental macroscopic traffic flow variables and characteristics, as well as a presentation of the fundamental relation of traffic flow theory.

Chapter 3 served as a literature review of computer simulation modelling fundamentals and best practices from the literature, in fulfilment of Dissertation Objective 1(d) in §1.3. The chapter opened with a presentation of the principles of simulation modelling in §3.1 and contained descriptions of the concepts and components of a simulation model, the different types of simulation models and four prevalent simulation modelling paradigms from the literature. In §3.2, the twelve steps typically followed in a simulation study were presented. The chapter closed in §3.3

with a discussion on the advantages and disadvantages associated with simulation modelling as well as its applicability to the modelling of road traffic.

A microscopic traffic simulation modelling framework built for the purpose of this dissertation was presented in Chapter 4, in fulfilment of Dissertation Objective 2(a) in §1.3. In §4.1, the design and implementation of the various components of the framework were described, including the road sections which comprise the road network, the vehicles which populate the road network and the traffic signals responsible for the control of traffic flow at intersections. In §4.2, the output generated by traffic simulation models implemented within the framework was described. The chapter closed in §4.3 with a description of the methods and procedure followed in the verification and validation of the simulation modelling framework.

Chapter 5 contained a literature review of existing traffic signal control techniques, in fulfilment of Dissertation Objectives 1(b) and 1(c) in §1.3. Reviews were presented of two fixed-time traffic signal control algorithms and two decentralised, self-organising traffic signal control algorithms from the literature. The first fixed-time traffic signal control algorithm (reviewed in §5.2.1) was presented in the Highway Capacity Manual [94], published by the Transportation Research Board, for equalising the degree of saturation along all intersection approach lanes. The second fixed-time traffic signal control algorithm (reviewed in §5.2.2) was proposed in [38] and attempts to facilitate the coordination between intersections which implement a fixed-time traffic signal control approach. The first self-organising traffic signal control algorithm reviewed in (§5.3.1) was that proposed by Gershenson and Rosenblueth in [38]. The second self-organising traffic signal control algorithm (reviewed in §5.3.2) was that of Lämmer and Helbing [53, 54]. The chapter closed with an appraisal of the two self-organising traffic signal control algorithms.

Three novel self-organising traffic signal control algorithms were introduced in Chapter 6. The first of these traffic signal control algorithms (presented in §6.1) was inspired by inventory control methodologies. The second algorithm (presented in §6.2) was inspired by the chemical process of osmosis, while the third algorithm (presented in §6.3) was a combination of the inventory and osmosis-inspired traffic control algorithms into a hybrid traffic control algorithm which was supplemented with an intersection utilisation maximisation technique.

The experimentation procedure followed for implementing and comparing the six different traffic signal control algorithms described in Chapters 5 and 6 was presented in Chapter 7. Each of the six algorithms were implemented for two different road network topologies, under a variety of prevailing traffic flow conditions, in fulfilment of Dissertation Objective 2 in §1.3. The first topology consisted of a corridor of four homogeneous intersections, while the second topology was a 3×4 gird of twelve homogeneous intersections. The chapter opened with a description of the experimental design which included a description and justification of the values of necessary model parameters and the performance measure indicators selected to evaluate and compare the algorithms, thereby achieving Dissertation Objective 2(b) in §1.3. In §7.2, the results of the various simulation runs were analysed and compared in fulfilment of Dissertation Objective 3 in §1.3, and the various traffic control algorithms were ranked according to their associated performances by means of the Tukey HSD method for comparing multiple systems, in accordance with Dissertation Objective 2(c) in §1.3. The chapter closed with an appraisal of each of the six traffic control algorithms investigated.

8.2 Appraisal of dissertation contributions

The main contributions of this dissertation are fourfold. The first contribution is the microscopic traffic simulation modelling framework developed. The framework is capable of facilitat-

ing accurate simulation modelling of real-world traffic flow by explicitly incorporating vehicle accelerations, vehicle turning and lane changing. Furthermore, the framework is customisable, allowing for the adjustment of vehicle arrival rates, speed limits, turning probabilities and rates of acceleration. The framework can accommodate simulation models which accurately replicate the data that are provided by radar detection sensors similar to those described in §1.2. Existing traffic signal control algorithms as well as customised traffic signal control algorithms may be implemented and investigated in the framework. These traffic signal control algorithms may utilise the data provided by the assumed radar detection sensors as input to inform traffic signal switching policies. The framework also allows for a variety of different road network topologies and intersection configurations to be implemented. During the execution of any simulation model implemented within the framework, a wealth of live, dynamic data are generated pertaining to the performances of traffic signal control algorithms and the state of the system in general. A paper describing this microscopic traffic simulation modelling framework has been published [29].

The second contribution of this dissertation is the introduction of three novel performance measure indicators for evaluating and comparing traffic signal control algorithms. The first of these novel performance measure indicators is the number of stops made by vehicles while travelling through the road network. This performance measure indicator allows one to gauge the effectiveness of a traffic signal control algorithm at facilitating coordination between intersections: the fewer the stops made by vehicles, the better the level of coordination achieved. The other two performance measures are the normalised delay time of vehicles and the normalised number of stops made by vehicles. The normalised delay time performance measure indicator provides an indication of the actual travel time of a vehicle in relation to its ideal travel time (and therefore factors in the desired speed and route choice of the vehicle) while the normalised number of stops performance measure indicator provides an indication of the number of stops a vehicle is likely to make in relation to the number of intersections it is expected to encounter.

The third contribution of this dissertation is the introduction of three novel decentralised, self-organising traffic signal control algorithms. Of these three algorithms, the algorithm inspired by osmosis and the hybrid algorithm (which harnesses both an inventory-inspired algorithm and the algorithm based on the process of osmosis) were shown to outperform three existing traffic signal control algorithms in the literature in terms of several different performance measure indicators for two different road network topologies, under varying traffic flow conditions. Furthermore, unlike the existing self-organising traffic signal control algorithms in the literature (described in §5.3.1 and §5.3.2), the effectiveness of which is determined by the selection of carefully chosen user-specified parameters, the three novel traffic control algorithms introduced rely instead solely on the prevailing traffic conditions, and the availability of physical space along the roadways to inform signal switching policies. A paper describing these three algorithms has been submitted for publication [12].

The final contribution of this dissertation is the comparison of both novel and existing self-organising traffic signal control algorithms in a realistic, microscopic traffic simulation modelling framework. From the literature reviewed, it is typical only to compare the performance a self-organising traffic signal control algorithm to that of a fixed-time traffic signal control algorithm, and in some cases to other vehicle-actuated traffic signal control algorithms under simplified, aggregated conditions. The results of these comparisons for a traffic corridor road network topology have been submitted for publication [12].

8.3 Future work

In this section, five suggestions are made with respect to possible future research pertaining to the design of self-organising traffic signal control algorithms and their associated performances as well as the microscopic traffic simulation modelling framework proposed in this dissertation.

Suggestion 8.3.1. Based on the results presented in §7.2.1 and §7.2.2, it was concluded that Hybrid was the best overall performing algorithm for the corridor network topology and that O-TSCA was the best performing algorithm for the grid network topology. The question is therefore posed: *Is it possible to alter either Hybrid or O-TSCA such that the resulting algorithm is able to outperform both Hybrid and O-TSCA at low, medium and high traffic flow conditions, for both road network topologies, while still remaining free of any parameters which may require adjusting or calibration?* One possible approach to answering this question would be to alter the gap acceptance threshold of the IUMSM within Hybrid so as to prevent it from terminating green times prematurely to arriving vehicle platoons which exhibit platoon dispersal. Alternatively, O-TSCA may be altered so that the algorithm re-evaluates the allocation of service after the last vehicle of each detected platoon clears the intersection, rather than waiting for the very last detected vehicle to clear the intersection. It is anticipated that this would improve the performance of O-TSCA under lighter prevailing traffic flow conditions.

Suggestion 8.3.2. In this dissertation, certain intersection approaches were grouped together in such a way that they receive service simultaneously. It is suggested that more flexible phase plans be investigated in which individual intersection approaches or even individual intersection approach lanes compete for service. It is also suggested that research take place into the effect on the various traffic signal control algorithms of heterogeneous intersection designs and heterogeneous intersection spacing within traffic network topologies.

Suggestion 8.3.3. It is desirable to compare the three novel traffic signal control algorithms introduced in this dissertation to the state-of-the-art algorithms which are currently implemented in real-world traffic networks. Such a comparison was not possible for this dissertation as the algorithmic logic of these state-of-the-art algorithms is not freely and readily available. Certain commercially available traffic simulation packages are, however, able to implement these algorithms in a black-box fashion. It is suggested, therefore, that attempts be made to implement the novel traffic signal control algorithms, proposed in this dissertation, in one of these commercially available traffic simulation packages so as to compare their performances to those algorithms which are currently considered to be the state-of-the-art.

Suggestion 8.3.4. It was observed that when there were relatively large numbers of vehicles present on the roadways of the traffic simulation models implemented within the microscopic traffic simulation modelling framework described in Chapter 4, the speed at which these simulation models ran was considerably reduced. This is due to the large amount of computer processing power required to continually iterate through vehicles, extracting, storing and manipulating the relevant data, so as to provide the necessary input data to the various traffic signal control algorithms. To overcome this problem of complexity, it is suggested that the distribution the traffic simulation models over several computers be considered so as to harness the processing power of numerous computing cores.

Suggestion 8.3.5. Pending the successful implementation of Suggestion 8.3.4 above, it is suggested that measures be investigated for improving upon the realism and validity of the

microscopic traffic simulation modelling framework presented in this dissertation. Such measures may include the incorporation of pedestrians into the framework, together with exclusive pedestrian crossing phases implemented at intersections, the incorporation of vehicle collisions, break-downs and other disruptions, vehicle overtaking manoeuvres, and the stochastic break-down of intersection traffic control signals. Following the successful implementation of these measures, it is suggested that the effect they have on the performances of traffic signal control algorithms be investigated.

References

[1] Allsop RE, 1972, *Delay at a fixed time traffic signal — I: Theoretical analysis*, Transportation Science, **6(3)**, pp. 260–285.

[2] Banks J, Carson JS, Nelson BL & Nicol DM, 2001, *Introduction to simulation*, pp. 3–22 in Fabrycky WJ & Mize JH (Eds), *Discrete-event system simulation*, Prentice-Hall, Upper Saddle River (NJ).

[3] Banks J, Carson JS, Nelson BL & Nicol DM, 2001, *Verification and validation of simulation models*, pp. 367–397 in Fabrycky WJ & Mize JH (Eds), *Discrete-event system simulation*, Prentice-Hall, Upper Saddle River (NJ).

[4] Banks J, 1998, *Comparing systems via simulation*, pp. 273–306 in Banks J (Ed), *Handbook of simulation: Principles, methodology, advances, applications, and practice*, Wiley-Interscience, New York (NY).

[5] Banks J, 1998, *Principles of simulation*, pp. 3–30 in Banks J (Ed), *Handbook of simulation: Principles, methodology, advances, applications, and practice*, Wiley-Interscience, New York (NY).

[6] Barceló J, 2010, *Models, traffic models, simulation and traffic simulation*, pp. 1–62 in Barceló J (Ed), *Fundamentals of traffic simulation*, Springer, New York (NY).

[7] Barth M & Boriboonsomsin K, 2008, *Real-world carbon dioxide impacts of traffic congestion*, Transportation Research Record: Journal of the Transportation Research Board, **2058(1)**, pp. 163–171.

[8] Bar-Yam Y, 1997, *Dynamics of complex systems*, Addison-Wesley, Reading (MA).

[9] Borshchev A & Filippov A, 2004, *From system dynamics and discrete event to practical agent based modeling: Reasons, techniques, tools*, Proceedings of the 22nd International Conference of the System Dynamics Society.

[10] Boxill SA & Yu L, 2000, *An evaluation of traffic simulation models for supporting ITS development*, Centre for Transportation Training and Research, Houston (TX).

[11] Brown TL, LeMay HE, Bursten BE, Murphy CJ & Woodward PM, 2012, *Chemistry: The central science*, Prentice Hall, Boston (MA).

[12] Burger AP, Einhorn MD & Van Vuuren JH, *Self-organising traffic control inspired by inventory theory and the process of osmosis*, Submitted.

[13] Burger AP, Einhorn MD & Van Vuuren JH, 2013, *The effects of incorporating vehicle acceleration explicitly into a microscopic traffic simulation model*, South African Journal of Industrial Engineering, **24(2)**, pp. 12–23.

[14] Cai C, Wong CK & Heydecker BG, 2009, *Adaptive traffic signal control using approximate dynamic programming*, Transportation Research Part C: Emerging Technologies, **17(5)**, pp. 456–474.

[15] Cassidy MJ, 1998, *Bivariate relations in nearly stationary highway traffic*, Transportation Research Part B: Methodological, **32(1)**, pp. 49–59.

[16] Chandler RE, Herman R & Montroll EW, 1958, *Traffic dynamics: Studies in car following*, Operations Research, **6(2)**, pp. 165–184.

[17] Coifman B, 2001, *Improved velocity estimation using single loop detectors*, Transportation Research Part A: Policy and Practice, **35(10)**, pp. 863–880.

[18] Crabtree M, Vincent R & Harrison S, 1996, *TRANSYT/10 user guide*, (Unpublished) Technical Report TRL Application Guide 28, Transportation Research Library, Berkshire.

[19] Craig DC, 1996, *Extensible hierarchical object-oriented logic simulation with an adaptable graphical user interface*, PhD thesis, Memorial University of Newfoundland, St Johns.

[20] Crutchfield JP, 1994, *The calculi of emergence: Computation, dynamics and induction*, Physica D: Nonlinear Phenomena, **75(1)**, pp. 11–54.

[21] Daganzo CF, 1997, *Fundamentals of transportation and traffic operations*, Pergamon-Elsevier, Oxford.

[22] De Wolf T & Holvoet T, 2005, *Emergence versus self-organisation: Different concepts but promising when combined*, pp. 1–15 in Bruekener SA, Seregundo GDM, Karageorgos A & Nagpal R (Eds), *Engineering self-organising systems: Methodologies and applications*, Springer, Berlin.

[23] Department of Transport Chief Directorate: National Roads, 1988, *Guidelines for the application of traffic signal phasing and control equipment*, (Unpublished) Technical Report 87, National Transport Commission, Pretoria.

[24] Dion F, Rakha H & Kang YS, 2004, *Comparison of delay estimates at under-saturated and over-saturated pre-timed signalized intersections*, Transportation Research Part B: Methodological, **38(2)**, pp. 99–122.

[25] Drew DR, 1968, *Traffic flow theory and control*, McGraw-Hill Book Company, New York, (NY).

[26] Edie LC, 1963, *Discussion of traffic flow measurements and definitions*, Proceedings of the 2nd International Symposium on the Theory of Traffic Flow, London, pp. 139–154.

[27] Einhorn MD, 2011, *An evaluation of the efficiency of self-organising versus fixed traffic signalling paradigms*, MSc Thesis, Stellenbosch University, Stellenbosch.

[28] Einhorn MD, Burger AP & van Vuuren JH, 2013, *An analytic testing framework for traffic control strategies*, Proceedings of the 42nd Annual Conference of the Operations Research Society of South Africa, pp. 139–148.

[29] Einhorn MD, Burger AP & van Vuuren JH, 2014, *Design of a detailed microscopic traffic simulation modelling framework*, Proceedings of the 43rd Annual Conference of the Operations Research Society of South Africa, pp. 1–8.

[30] Fellendorf M & Vortisch P, 2010, *Microscopic traffic flow simulator VISSIM*, pp. 63–94 in Barceló J (Ed), *Fundamentals of traffic simulation*, Springer, New York (NY).

[31] Fellendorf M, 1994, *VISSIM: A microscopic simulation tool to evaluate actuated signal control including bus priority*, Proceedings of the 64th Institute of Transportation Engineers Annual Meeting, pp. 1–9.

[32] Forbes T & Simpson ME, 1968, *Driver-and-vehicle response in freeway deceleration waves*, Transportation Science, **2(1)**, pp. 77–104.

[33] Gartner NH, Little JD & Gabbay H, 1975, *Optimization of traffic signal settings by mixed-integer linear programming. Part I: The network coordination problem*, Transportation Science, **9(4)**, pp. 321–343.

[34] Gartner NH, Little JD & Gabbay H, 1975, *Optimization of traffic signal settings by mixed-integer linear programming. Part II: The network synchronisation problem*, Transportation Science, **9(4)**, pp. 344–366.

[35] Gazis DC, Herman R & Potts RB, 1959, *Car-following theory of steady-state traffic flow*, Operations Research, **7(4)**, pp. 499–505.

[36] Gazis DC, Herman R & Rothery RW, 1961, *Nonlinear follow-the-leader models of traffic flow*, Operations Research, **9(4)**, pp. 545–567.

[37] Gershenson C, 2004, *Self-organizing traffic lights*, arXiv preprint nlin/0411066.

[38] Gershenson C & Rosenblueth DA, 2012, *Adaptive self-organization vs static optimization: A qualitative comparison in traffic light coordination*, Kybernetes, **41(3/4)**, pp. 386–403.

[39] Gibson D, 1981, *Available computer models for traffic operations analysis*, (Unpublished) Technical Report FHWA-TS-82-207, Transportation Research Board, Washington (DC).

[40] Goodwin P, 2004, *The economic costs of road traffic congestion*, Discussion Paper, The Rail Freight Group, Transport Studies Unit, University College, London.

[41] Greenshields BD, Shapiro D & Ericksen E, 1946, *Traffic performance at urban street intersections*, (Unpublished) Technical Report 1, Bureau of Highway Traffic, Yale University, New Haven (CT).

[42] Gyimesi K, Vincent C & Lamba N, 2011, *Frustration rising: IBM 2011 commuter pain survey*, [Online], [Cited April 23rd, 2014], Available from `http://www.ibm.com/press/us/en/presskit/35314.wss`.

[43] Hall M & Willumsen L, 1980, *SATURN-a simulation-assignment model for the evaluation of traffic management schemes*, Traffic Engineering & Control, **21(4)**, pp. 169–170.

[44] Helbing D, Lämmer S & Lebacque JP, 2005, *Self-organized control of irregular or perturbed network traffic*, pp. 239–274 in Deissenberg C & Hartl RF (Eds), *Optimal control and dynamic games*, Springer, New York (NY).

[45] Henry JJ, Farges JL & Tuffal J, 1984, *The PRODYN real time traffic algorithm*, Proceedings of the 4th IFAC/ IFIP/ IFORS Conference on Control in Transportation Systems, pp. 307–311.

[46] Herman R, Montroll EW, Potts RB & Rothery RW, 1959, *Traffic dynamics: Analysis of stability in car following*, Operations Research, **7(1)**, pp. 86–106.

[47] Heyligen F, 2001, *The science of self-organisation and adaptivity*, EOLSS Publishers, Oxford.

[48] Hoogendoorn SP & Bovy PH, 2001, *State-of-the-art of vehicular traffic flow modelling*, Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, **215(4)**, pp. 283–303.

[49] Hoogendoorn S & Knoop V, 2013, *Traffic flow theory and modelling*, pp. 125–159 in van Wee B, Annema JA & Bannister D (Eds), *The transport system and transport policy: An introduction*, Edward Elgar Publishing Limited, Cheltenham.

[50] Hunt PB, Robertson DI, Bretherton RD & Winton RI, 1981, *SCOOT: A traffic responsive method of coordinating signals*, (Unpublished) Technical Report 1041, Transport and Road Research Laboratory, Crowthorne, Berkshire.

[51] Klein LA, Mills MK & Gibson DR, 2006, *Traffic detector handbook: Volume II*, (Unpublished) Technical Report FHWA-HRT-06-108, Federal Highway Administration, Washington (DC).

[52] Lämmer S, Donner R & Helbing D, 2008, *Anticipative control of switched queueing systems*, The European Physical Journal B-Condensed Matter and Complex Systems, **63(3)**, pp. 341–347.

[53] Lämmer S & Helbing D, 2008, *Self-control of traffic lights and vehicle flows in urban road networks*, Journal of Statistical Mechanics: Theory and Experiment, **2008(04)**, pp. 1–34 (P04019).

[54] Lämmer S & Helbing D, 2010, *Self-stabilizing decentralized signal control of realistic, saturated network traffic*, (Working paper 10-09-019), Santa Fe Institute, Santa Fe (CA).

[55] Law AM, 2000, *Building valid, credible, and appropriately detailed simulation models*, pp. 243–276 in Law AM (Ed), *Simulation modelling and analysis*, McGraw-Hill, Boston (MA).

[56] Law AM, 2000, *Output data analysis for a single system*, pp. 485–547 in Law AM (Ed), *Simulation modelling and analysis*, McGraw-Hill, Boston (MA).

[57] Leo CJ & Pretty RL, 1992, *Numerical simulation of macroscopic continuum traffic models*, Transportation Research Part B: Methodological, **26(3)**, pp. 207–220.

[58] Liao LC, 1998, *A review of the optimized policies for adaptive control strategy (OPAC)*, California PATH Program, Institute of Transportation Studies, University of California, Berkley (CA).

[59] Lighthill MJ & Whitham GB, 1955, *On kinematic waves. II. A theory of traffic flow on long crowded roads*, Proceedings of the Royal Society of London, Series A: Mathematical and Physical Sciences, **229(1178)**, pp. 317–345.

[60] Little JD, 1966, *The synchronization of traffic signals by mixed-integer linear programming*, Operations Research, **14(4)**, pp. 568–594.

[61] Logghe S, 2003, *Dynamic modeling of heterogeneous vehicular traffic*, PhD Thesis, Katholieke Universiteit Leuven, Leuven.

[62] Lowrie P, 1982, *The Sydney coordinated adaptive traffic system — Principles, methodology, algorithms*, Proceedings of the 207th International Conference on Road Traffic Signalling, 1982, London.

[63] Maerivoet S & De Moor B, 2005, *Traffic flow theory*, arXiv preprint physics/0507126.

[64] Mauro V & Di Taranto C, 1990, *Utopia*, Proceedings of the 6th IFAC/ IFIP/ IFORS Symposium on Control and Communication in Transportation, Paris, pp. 245–252.

[65] May AD, 1990, *Traffic flow fundamentals*, Prentice-Hall, Englewood Cliffs (NJ).

[66] McMillan C, Gonzalez RF & Schriber TJ, 1973, *Systems analysis: A computer approach to decision models*, Richard D. Irwin, Inc., Homewood (IL).

[67] Michalopoulos PG, Yi P & Lyrintzis AS, 1993, *Continuum modelling of traffic dynamics for congested freeways*, Transportation Research Part B: Methodological, **27(4)**, pp. 315–332.

[68] Miller AJ, 1963, *Settings for fixed-cycle traffic signals*, Operations Research, **14(4)**, pp. 373–386.

[69] Mimbela LEY & Klein LA, 2003, *Summary of vehicle detection and surveillance technologies used in intelligent transportation systems*, Vehicle Detector Clearinghouse, Southwest Technology Development Institute, New Mexico State University, Las Cruces (NM).

[70] Moustefaoui S, Rana O, Foukia N, Hassas S, Marzo G, Aart C & Karageorgos A, 2003, *Self-organising applications: A survey*, Proceedings of the International Workshop on Engineering Self-Organising Applications, pp. 62–69.

[71] Musselman KJ, 1998, *Guidelines for success*, pp. 721–744 in Banks J (Ed), *Handbook of simulation: Principles, methodology, advances, applications, and practice*, Wiley-Interscience, New York (NY).

[72] National Roads Act of 1998, *Tariffs for different categories or road users and classes of motor vehicles*, South Africa [Statute], Government Gazette, **36912**, Pretoria.

[73] Naylor TH, Balintfy JL, Burdick DS & Chu K, 1968, *Computer simulation techniques*, John Wiley and Sons, Inc., New York (NY).

[74] Newell GF, 1965, *Approximation methods for queues with application to the fixed-cycle traffic light*, SIAM Review, **7(2)**, pp. 223–240.

[75] Odell J, 2002, *Agents and complex systems*, Journal of Object Technology, **1(2)**, pp. 35–45.

[76] *Oxford English Dictionary*, 2005, Oxford University Press, Oxford.

[77] Papacostas CS & Prevedouros PD, 2001, *Capacity and level of service analysis*, pp. 133–230 in Curless L (Ed), *Transportation engineering and planning*, Prentice-Hall, Upper Saddle River (NJ).

[78] Papacostas CS & Prevedouros PD, 2001, *Transportation software*, pp. 626–652 in Curless L (Ed), *Transportation engineering and planning*, Prentice-Hall, Upper Saddle River (NJ).

[79] Papadimitriou CH & Tsitsiklis JN, 1999, *The complexity of optimal queuing network control*, Mathematics of Operations Research, **24(2)**, pp. 293–305.

[80] Pegden CD, Sadowski RP & Shannon RE, 1995, *Introduction to simulation using SIMAN*, McGraw-Hill, Inc., Boston (MA).

[81] Perkins JR & Kumar P, 1989, *Stable, distributed, real-time scheduling of flexible manufacturing/assembly/diassembly systems*, IEEE Transactions on Automatic Control, **34(2)**, pp. 139–148.

[82] Pipes LA, 1953, *An operational analysis of traffic dynamics*, Journal of Applied Physics, **24(3)**, pp. 274–281.

[83] Porche I & Lafortune S, 1997, *Dynamic traffic control: Decentralized and coordinated methods*, Proceedings of the IEEE Conference on Intelligent Transportation System, pp. 930–935.

[84] Richards PI, 1956, *Shock waves on the highway*, Operations Research, **4(1)**, pp. 42–51.

[85] Rosenblueth DA & Gershenson C, 2011, *A model of city traffic based on elementary cellular automata*, Complex Systems, **19(4)**, pp. 305.

[86] SCATS, 2014, *SCATS — The benchmark in urban traffic control*, [Online], [Cited June 19th, 2014], Available from `http://www.scats.com.au/`.

[87] SCOOT, 2014, *SCOOT — The world's leading adaptive traffic control system*, [Online], [Cited June 19th, 2014], Available from `http://www.scoot-utc.com/`.

[88] *Self-organising systems FAQ*, 2014, [Online], [Cited October 31st, 2014], Available from `http://www.calresco.org/sos/sosfaq.htm`.

[89] Starr C & Taggart R, 1989, *Biology: The unity and diversity of life*, Wandsworth Publishing Company, Belmont (CA).

[90] Sterman JD, 2000, *Business dynamics: Systems thinking and modeling for a complex world*, Irwin/McGraw-Hill, Boston (MA).

[91] Sweet M, 2013, *Traffic congestion's economic impacts: Evidence from US metropolitan regions*, Urban Studies, **50(15)**, pp. 1–23.

[92] The Telegraph, 2012, *Traffic congestion costs UK economy £4.3bn a year*, [Online], [Cited April 23rd, 2014], Available from `http://www.telegraph.co.uk`.

[93] Todosiev E & Barbosa L, 1964, *A proposed model for the driver-vehicle system*, Traffic Engineering, **34(17)**, pp. 17–20.

[94] Transportation Research Board, 2000, *Highway capacity manual*, Washington (DC).

[95] Van der Merwe E, 2009, *Improving the flow of an isolated traffic system*, Fourth Year Engineering Thesis, Stellenbosch University, Stellenbosch.

[96] Wardrop JG, 1952, *Some theoretical aspects of road traffic research*, Proceedings of the Institution of Civil Engineers, pp. 325–378.

[97] Wavetronix, 2014, *Wavetronix SmartSensor Advance*, [Online], [Cited June 19th, 2014], Available from `http://www.wavetronix.com/en/products/smartsensor/advance`.

[98] Wavetronix, 2014, *Wavetronix.com*, [Online], [Cited June 19th, 2014], Available from `http://www.wavetronix.com`.

[99] Webster FV, 1958, *Traffic signal settings*, Road Research Technical paper No. 39, Road Research Laboratory, Berkshire.

[100] Winston WL, 2004, *Deterministic EOQ inventory models*, pp. 846–879 in Winston WL (Ed), *Operations research — Applications and algorithms*, Thomson Learning, Belmont (CA).

[101] XJ Technologies, 2014, *AnyLogic – Multimethod simulation software*, [Online], [Cited June 19th, 2014], Available from: `http://www.anylogic.com/`, [Online], [Cited September 24th,

[102] XJ Technologies, 2014, *AnyLogic – Online help*, [Online], [Cited September 24th, 2014], Available from `http://www.anylogic.com/anylogic/help/`.

[103] Zubillaga D, Cruz G, Aguilar LD, Zapotécatl J, Fernández N, Aguilar J, Rosenblueth DA & Gershenson C, 2014, *Measuring the complexity of self-organizing traffic lights*, Entropy, **16(5)**, pp. 2384–2407.

APPENDIX A

Input data for the Adam Tas Road and Bird Street intersection

This appendix contains the observed vehicle proportions for each of the approaches to the Adam Tas Road & Bird Street intersection, as well as samples of the observed green times from the morning-peak, midday, and afternoon-peak traffic flow periods which were used for the validation of the microscopic traffic simulation modelling framework in Chapter 4.

A.1 Vehicle movement proportions

This section contains the observed vehicle proportions for each of the approaches to the Adam Tas Road & Bird Street intersection in Stellenbosch, South Africa.

Table A.1: *Turning proportions of vehicles approaching along Adam Tas Road (West).*

Vehicles approaching along Adam Tas Road (West)							
		Total number of vehicles that:			% of vehicles that:		
Time	Total number of vehicles	turn left	travel straight	turn right	turn left	travel straight	turn right
06:30–06:45	155	48	105	2	31	68	1
06:45–07:00	161	60	101	0	37	63	0
07:00–07:15	256	90	165	1	35	64	0
07:15–07:30	193	57	132	4	30	68	2
07:30–07:45	155	30	122	3	19	79	2
07:45–08:00	150	36	112	2	24	75	1
08:00–08:15	226	65	155	6	29	69	3
08:15–08:30	159	63	94	2	40	59	1
08:30–08:45	137	41	91	5	30	66	4
08:45–09:00	166	63	97	6	38	58	4
09:00–09:15	160	62	95	3	39	59	2
09:15–09:30	154	57	94	3	37	61	2
09:30–09:45	164	52	103	9	32	63	5
09:45–10:00	151	52	98	1	34	65	1
10:00–10:15	136	49	85	2	36	63	1

Continued on next page

Table A.1 – continued from previous page

Vehicles approaching along Adam Tas Road (West)							
		Total number of vehicles that:			% of vehicles that:		
Time	Total number of vehicles	turn left	travel straight	turn right	turn left	travel straight	turn right
10:15–10:30	147	47	93	7	32	63	5
10:30–10:45	172	47	122	3	27	71	2
10:45–11:00	120	44	74	2	37	62	2
11:00–11:15	171	55	111	5	32	65	3
11:15–11:30	173	68	103	2	39	60	1
11:30–11:45	122	38	81	3	31	66	2
11:45–12:00	158	53	102	3	34	65	2
12:00–12:15	205	73	123	9	36	60	4
12:15–12:30	121	48	66	7	40	55	6
12:30–12:45	152	52	96	4	34	63	3
12:45–13:00	164	52	107	5	32	65	3
13:00–13:15	198	76	114	8	38	58	4
13:15–13:30	154	53	97	4	34	63	3
13:30–13:45	156	38	108	10	24	69	6
13:45–14:00	173	72	98	3	42	57	2
14:00–14:15	132	52	77	3	39	58	2
14:15–14:30	184	58	120	6	32	65	3
14:30–14:45	211	65	144	2	31	68	1
14:45–15:00	166	52	109	5	31	66	3
15:00–15:15	144	53	90	1	37	63	1
15:15–15:30	200	74	121	5	37	61	3
15:30–15:45	152	49	99	4	32	65	3
15:45–16:00	202	80	107	15	40	53	7
16:00–16:15	100	32	67	1	32	67	1
16:15–16:30	185	76	105	4	41	57	2
16:30–16:45	194	64	125	5	33	64	3
16:45–17:00	248	80	157	11	32	63	4
17:00–17:15	175	55	120	0	31	69	0
17:15–17:30	194	66	124	4	34	64	2
17:30–17:45	279	90	187	2	32	67	1
17:45–18:00	206	72	132	2	35	64	1

Table A.2: *Turning proportions of vehicles approaching along Bird Street (South).*

Vehicles approaching along Bird Street (South)							
		Total number of vehicles that:			% of vehicles that:		
Time	Total number of vehicles	turn left	travel straight	turn right	turn left	travel straight	turn right
06:30–06:45	44	1	31	12	2	70	27
06:45–07:00	77	2	57	18	3	74	23
07:00–07:15	101	0	62	39	0	61	39

Continued on next page

Table A.2 – continued from previous page

Vehicles approaching along Bird Street (South)							
		Total number of vehicles that:			% of vehicles that:		
Time	Total number of vehicles	turn left	travel straight	turn right	turn left	travel straight	turn right
07:15–07:30	98	2	66	30	2	67	31
07:30–07:45	133	3	77	53	2	58	40
07:45–08:00	156	3	87	66	2	56	42
08:00–08:15	111	4	63	44	4	57	40
08:15–08:30	137	1	96	40	1	70	29
08:30–08:45	121	4	74	43	3	61	36
08:45–09:00	117	8	70	39	7	60	33
09:00–09:15	112	2	70	40	2	63	36
09:15–09:30	119	6	72	41	5	61	34
09:30–09:45	88	3	51	34	3	58	39
09:45–10:00	120	3	79	38	3	66	32
10:00–10:15	112	3	75	34	3	67	30
10:15–10:30	162	1	106	55	1	65	34
10:30–10:45	106	2	68	36	2	64	34
10:45–11:00	116	6	71	39	5	61	34
11:00–11:15	128	4	78	46	3	61	36
11:15–11:30	147	3	87	57	2	59	39
11:30–11:45	85	3	60	22	4	71	26
11:45–12:00	136	4	81	51	3	60	38
12:00–12:15	171	6	114	51	4	67	30
12:15–12:30	130	1	72	57	1	55	44
12:30–12:45	116	2	75	39	2	65	34
12:45–13:00	159	7	100	52	4	63	33
13:00–13:15	185	9	110	66	5	59	36
13:15–13:30	153	8	88	57	5	58	37
13:30–13:45	149	5	95	49	3	64	33
13:45–14:00	170	6	98	66	4	58	39
14:00–14:15	111	5	79	27	5	71	24
14:15–14:30	168	3	96	69	2	57	41
14:30–14:45	178	2	111	65	1	62	37
14:45–15:00	136	2	86	48	1	63	35
15:00–15:15	156	6	105	45	4	67	29
15:15–15:30	178	9	108	61	5	61	34
15:30–15:45	178	8	114	56	4	64	31
15:45–16:00	168	6	121	41	4	72	24
16:00–16:15	151	4	99	48	3	66	32
16:15–16:30	195	2	131	62	1	67	32
16:30–16:45	227	0	135	92	0	59	41
16:45–17:00	269	7	155	107	3	58	40
17:00–17:15	243	2	145	96	1	60	40
17:15–17:30	246	1	148	97	0	60	39
17:30–17:45	212	2	116	94	1	55	44
17:45–18:00	215	2	120	93	1	56	43

Table A.3: *Turning proportions of vehicles approaching along Adam Tas Road (East).*

Vehicles approaching along Adam Tas Road (East)							
		Total number of vehicles that:			% of vehicles that:		
Time	Total number of vehicles	turn left	travel straight	turn right	turn left	travel straight	turn right
06:30–06:45	121	18	88	15	15	73	12
06:45–07:00	201	37	126	38	18	63	19
07:00–07:15	364	88	233	43	24	64	12
07:15–07:30	262	52	178	32	20	68	12
07:30–07:45	459	151	254	54	33	55	12
07:45–08:00	243	78	127	38	32	52	16
08:00–08:15	260	77	150	33	30	58	13
08:15–08:30	210	54	130	26	26	62	12
08:30–08:45	262	76	153	33	29	58	13
08:45–09:00	90	35	44	11	39	49	12
09:00–09:15	180	56	97	27	31	54	15
09:15–09:30	120	40	63	17	33	53	14
09:30–09:45	179	56	108	15	31	60	8
09:45–10:00	100	36	53	11	36	53	11
10:00–10:15	161	45	83	33	28	52	20
10:15–10:30	160	64	76	20	40	48	13
10:30–10:45	164	57	87	20	35	53	12
10:45–11:00	102	33	54	15	32	53	15
11:00–11:15	144	41	77	26	28	53	18
11:15–11:30	188	59	104	25	31	55	13
11:30–11:45	154	57	77	20	37	50	13
11:45–12:00	169	45	98	26	27	58	15
12:00–12:15	171	60	85	26	35	50	15
12:15–12:30	123	36	71	16	29	58	13
12:30–12:45	177	61	94	22	34	53	12
12:45–13:00	136	39	79	18	29	58	13
13:00–13:15	189	48	117	24	25	62	13
13:15–13:30	175	54	102	19	31	58	11
13:30–13:45	206	61	125	20	30	61	10
13:45–14:00	106	34	61	11	32	58	10
14:00–14:15	185	47	99	39	25	54	21
14:15–14:30	163	47	85	31	29	52	19
14:30–14:45	190	50	120	20	26	63	11
14:45–15:00	167	55	91	21	33	54	13
15:00–15:15	164	52	95	17	32	58	10
15:15–15:30	199	41	119	39	21	60	20
15:30–15:45	128	31	78	19	24	61	15
15:45–16:00	211	59	125	27	28	59	13
16:00–16:15	159	51	86	22	32	54	14
16:15–16:30	239	55	132	52	23	55	22
16:30–16:45	211	60	119	32	28	56	15

Continued on next page

Table A.3 – continued from previous page

Vehicles approaching along Adam Tas Road (East)							
		Total number of vehicles that:			% of vehicles that:		
Time	Total number of vehicles	turn left	travel straight	turn right	turn left	travel straight	turn right
16:45–17:00	176	64	89	23	36	51	13
17:00–17:15	202	84	96	22	42	48	11
17:15–17:30	186	71	83	32	38	45	17
17:30–17:45	206	77	109	20	37	53	10
17:45–18:00	188	67	95	26	36	51	14

Table A.4: *Turning proportions of vehicles approaching along Bird Street (North).*

Vehicles approaching along Bird Street (North)							
		Total number of vehicles that:			% of vehicles that:		
Time	Total number of vehicles	turn left	travel straight	turn right	turn left	travel straight	turn right
06:30–06:45	157	8	74	75	5	47	48
06:45–07:00	207	8	105	94	4	51	45
07:00–07:15	241	9	141	91	4	59	38
07:15–07:30	220	6	127	87	3	58	40
07:30–07:45	208	12	126	70	6	61	34
07:45–08:00	211	11	135	65	5	64	31
08:00–08:15	252	10	148	94	4	59	37
08:15–08:30	230	13	126	91	6	55	40
08:30–08:45	168	16	104	48	10	62	29
08:45–09:00	207	10	135	62	5	65	30
09:00–09:15	233	15	106	112	6	45	48
09:15–09:30	143	15	109	19	10	76	13
09:30–09:45	175	11	96	68	6	55	39
09:45–10:00	160	8	103	49	5	64	31
10:00–10:15	183	17	103	63	9	56	34
10:15–10:30	169	18	91	60	11	54	36
10:30–10:45	177	16	99	62	9	56	35
10:45–11:00	156	16	85	55	10	54	35
11:00–11:15	134	17	76	41	13	57	31
11:15–11:30	168	13	95	60	8	57	36
11:30–11:45	179	19	84	76	11	47	42
11:45–12:00	191	27	90	74	14	47	39
12:00–12:15	142	22	69	51	15	49	36
12:15–12:30	155	10	87	58	6	56	37
12:30–12:45	169	14	86	69	8	51	41
12:45–13:00	139	15	68	56	11	49	40
13:00–13:15	203	14	121	68	7	60	33
13:15–13:30	222	17	130	75	8	59	34
13:30–13:45	107	9	67	31	8	63	29

Continued on next page

Table A.4 – continued from previous page

Vehicles approaching along Bird Street (North)							
		Total number of vehicles that:			% of vehicles that:		
Time	Total number of vehicles	turn left	travel straight	turn right	turn left	travel straight	turn right
13:45–14:00	100	4	58	38	4	58	38
14:00–14:15	127	8	63	56	6	50	44
14:15–14:30	135	4	69	62	3	51	46
14:30–14:45	164	3	107	54	2	65	33
14:45–15:00	171	19	86	66	11	50	39
15:00–15:15	159	16	87	56	10	55	35
15:15–15:30	140	14	86	40	10	61	29
15:30–15:45	152	7	86	59	5	57	39
15:45–16:00	159	12	96	51	8	60	32
16:00–16:15	155	13	82	60	8	53	39
16:15–16:30	159	10	95	54	6	60	34
16:30–16:45	170	5	119	46	3	70	27
16:45–17:00	233	13	166	54	6	71	23
17:00–17:15	188	15	117	56	8	62	30
17:15–17:30	208	21	139	48	10	67	23
17:30–17:45	196	16	128	52	8	65	27
17:45–18:00	227	17	166	44	7	73	19

A.2 Individual phase green times

Samples of the green times observed during the morning-peak, midday-peak and and afternoon-peak traffic flow periods for each of the four phases of the traffic signal cycle at the Adam Tas Road & Bird Street intersection are presented in this section. The means of these observed green times were used as representative values for the green times implemented during the validation process of the microscopic traffic simulation modelling framework described in Chapter 4.

Table A.5: *Observed morning-peak green times (in seconds) at the Adam Tas Road & Bird Street intersection.*

Time of day	Phase 1	Phase 2	Phase 3	Phase 4
Morning-peak period				
07:58	13	50	51	17
08:02	10	48	65	18
08:04	10	53	70	11
08:04	11	60	34	19
08:06	8	55	60	17
08:07	10	45	60	17
08:09	10	56	40	16
08:10	8	64	50	16
08:12	10	70	53	13

Continued on next page

Table A.5 – continued from previous page

Time of day	Phase 1	Phase 2	Phase 3	Phase 4
08:14	7	70	40	15
08:16	10	58	58	4
08:18	11	48	76	14
08:19	11	50	54	15
08:21	11	45	70	14
Mean	10.00	55.14	55.79	14.71
Midday period				
10:43	12	43	65	15
10:44	15	57	41	15
10:45	14	62	50	15
10:47	12	50	75	16
10:48	10	45	60	17
10:50	13	47	84	0
10:52	13	38	90	0
10:53	10	32	86	0
Mean	12.38	46.75	68.88	9.75
Afternoon-peak period				
17:18	7	48	80	0
17:21	9	60	48	14
17:23	0	65	56	14
17:26	12	51	65	14
17:28	12	40	70	14
17:31	12	60	80	0
17:33	10	45	77	0
17:35	9	46	75	0
17:38	0	70	70	0
17:40	9	42	75	16
17:43	10	60	38	0
17:45	0	70	80	0
17:48	0	52	80	0
17:50	12	56	65	0
17:52	12	65	60	0
17:55	10	57	80	0
17:57	9	45	90	0
18:00	11	42	75	0
18:02	10	30	80	0
Mean	8.11	52.84	70.74	3.79

APPENDIX B

Algorithmic comparison and ranking results

The ranking and performances of the six traffic control algorithms investigated in Chapter 7 are presented in this appendix in respect of the five performance measure indicators considered. The results for the corridor network topology are presented in §B.1, while the results for the grid network topology are presented in §B.2. The tables presented in this chapter compare the six traffic control algorithms and provide information pertaining to the magnitude by which one algorithm may be expected to outperform another a 95% confidence level.

B.1 Algorithm rankings for the corridor network topology

The results contained in the tables presented in this section correspond to the results shown in Figures 7.5, 7.6 and 7.7 for low, medium and high average vehicle arrival rates to the corridor network, respectively.

Traffic control algorithm		Average Improvement		Interval limit		Interval limit	
1	2	(s)	(%)	Lower	Upper	Lower (%)	Upper (%)
Hybrid	Gersh	0.78	6.2%	0.52	1.04	4.1 %	8.2 %
Hybrid	I-TSCA	2.19	15.6%	1.93	2.45	13.7%	17.4%
Hybrid	LH	4.33	26.7%	4.07	4.59	25.1%	28.3%
Hybrid	O-TSCA	5.36	31.1%	5.10	5.62	29.6%	32.6%
Hybrid	Fixed	5.59	32.0%	5.33	5.85	30.5%	33.5%
Gersh	I-TSCA	1.41	10.0%	1.15	1.67	8.2%	11.9%
Gersh	LH	3.55	21.9%	3.29	3.81	20.3%	23.5%
Gersh	O-TSCA	4.58	26.5%	4.32	4.84	25.0%	28.1%
Gersh	Fixed	4.81	27.5%	4.55	5.07	26.0%	29.0%
I-TSCA	LH	2.14	13.2%	1.88	2.40	11.6%	14.8%
I-TSCA	O-TSCA	3.17	18.4%	2.91	3.43	16.9%	19.9%
I-TSCA	Fixed	3.40	19.4%	3.14	3.66	17.9%	20.9%
LH	O-TSCA	1.03	6.0%	0.77	1.29	4.5%	7.5%
O-TSCA	Fixed	0.23	1.3%	−0.03	0.49	−0.2%	2.8%

Table B.1: *Algorithmic comparison and ranking with respect to the mean delay time for the scenario in which* $\lambda = 10$ *for the corridor network topology.*

Traffic control algorithm		Average Improvement		Interval limit		Interval limit	
1	2		(%)	Lower	Upper	Lower (%)	Upper (%)
Hybrid	Gersh	0.01	0.9 %	0.01	0.01	0.6 %	1.2 %
Hybrid	I-TSCA	0.05	3.7 %	0.04	0.05	3.4 %	4.0 %
Hybrid	LH	0.06	4.6 %	0.05	0.06	4.3 %	4.9 %
Hybrid	Fixed	0.09	7.0 %	0.09	0.09	6.7 %	7.3 %
Hybrid	O-TSCA	0.10	7.9 %	0.10	0.10	7.6 %	8.2 %
Gersh	I-TSCA	0.03	2.8 %	0.03	0.04	2.5 %	3.1 %
Gersh	LH	0.05	3.8 %	0.04	0.05	3.5 %	4.1 %
Gersh	Fixed	0.08	6.2 %	0.07	0.08	5.9 %	6.5 %
Gersh	O-TSCA	0.09	7.1 %	0.09	0.09	6.8 %	7.4 %
I-TSCA	LH	0.01	1.0 %	0.01	0.02	0.7 %	1.3 %
I-TSCA	Fixed	0.04	3.5 %	0.04	0.05	3.2 %	3.8 %
I-TSCA	O-TSCA	0.06	4.4 %	0.05	0.06	4.1 %	4.7 %
LH	Fixed	0.03	2.5 %	0.03	0.04	2.2 %	2.8 %
Fixed	O-TSCA	0.01	0.9 %	0.01	0.02	0.6 %	1.2 %

Table B.2: *Algorithmic comparison and ranking with respect to the mean normalised delay time for the scenario in which $\lambda = 10$ for the corridor network topology.*

Traffic control algorithm		Average Improvement		Interval limit		Interval limit	
1	2	(s)	(%)	Lower	Upper	Lower (%)	Upper (%)
Hybrid	Gersh	1.80	2.0 %	−5.66	9.27	−6.2 %	10.2 %
Hybrid	I-TSCA	6.93	7.2 %	−0.54	14.40	−0.6 %	15.0 %
Hybrid	O-TSCA	19.14	17.7 %	11.67	26.61	10.8 %	24.6 %
Hybrid	Fixed	28.67	24.4 %	21.21	36.14	18.0 %	30.7 %
Hybrid	LH	39.36	30.7 %	31.89	46.82	24.8 %	36.5 %
Gersh	I-TSCA	5.13	5.3 %	−2.34	12.60	−2.4 %	13.1 %
Gersh	O-TSCA	17.34	16.0 %	9.87	24.81	9.1 %	22.9 %
Gersh	Fixed	26.87	22.8 %	19.40	34.34	16.5 %	29.2 %
Gersh	LH	37.55	29.3 %	30.08	45.02	23.4 %	35.1 %
I-TSCA	O-TSCA	12.21	11.3 %	4.74	19.68	4.4 %	18.2 %
I-TSCA	Fixed	21.74	18.5 %	14.27	29.21	12.1 %	24.8 %
I-TSCA	LH	32.42	25.3 %	24.96	39.89	19.4 %	31.1 %
O-TSCA	Fixed	9.53	8.1 %	2.06	17.00	1.8 %	14.4 %
Fixed	LH	10.68	8.3 %	3.21	18.15	2.5 %	14.1 %

Table B.3: *Algorithmic comparison and ranking with respect to the maximum delay time for the scenario in which $\lambda = 10$ for the corridor network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
O-TSCA	LH	0.07	7.5 %	0.05	0.09	5.6 %	9.3 %
O-TSCA	Hybrid	0.13	12.8 %	0.11	0.15	11.1 %	14.6 %
O-TSCA	Gersh	0.31	26.3 %	0.29	0.33	24.8 %	27.8 %
O-TSCA	I-TSCA	0.37	29.6 %	0.35	0.38	28.2 %	31.0 %
O-TSCA	Fixed	0.98	52.8 %	0.96	0.99	51.9 %	53.8 %
LH	Hybrid	0.06	5.8 %	0.04	0.08	4.0 %	7.6 %
LH	Gersh	0.24	20.4 %	0.22	0.26	18.9 %	21.9 %
LH	I-TSCA	0.30	23.9 %	0.28	0.31	22.5 %	25.4 %
LH	Fixed	0.91	49.0 %	0.89	0.92	48.1 %	50.0 %
Hybrid	Gersh	0.18	15.5 %	0.17	0.20	14.0 %	17.0 %
Hybrid	I-TSCA	0.24	19.2 %	0.22	0.26	17.8 %	20.7 %
Hybrid	Fixed	0.85	45.9 %	0.83	0.87	44.9 %	46.8 %
Gersh	I-TSCA	0.06	4.5 %	0.04	0.07	3.0 %	5.9 %
I-TSCA	Fixed	0.61	33.0 %	0.59	0.63	32.0 %	34.0 %

Table B.4: *Algorithmic comparison and ranking with respect to the mean number of stops made by vehicles for the scenario in which $\lambda = 10$ for the corridor network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
LH	O-TSCA	0.02	3.4 %	0.01	0.03	1.7 %	5.1 %
LH	Hybrid	0.05	8.0 %	0.04	0.06	6.4 %	9.6 %
LH	Gersh	0.15	21.4 %	0.14	0.16	20.0 %	22.8 %
LH	I-TSCA	0.24	30.1 %	0.23	0.25	28.9 %	31.4 %
LH	Fixed	0.58	50.6 %	0.57	0.59	49.7 %	51.4 %
O-TSCA	Hybrid	0.03	4.8 %	0.02	0.04	3.2 %	6.4 %
O-TSCA	Gersh	0.13	18.6 %	0.12	0.14	17.2 %	20.0 %
O-TSCA	I-TSCA	0.22	27.7 %	0.21	0.23	26.5 %	28.9 %
O-TSCA	Fixed	0.56	48.8 %	0.55	0.57	48.0 %	49.7 %
Hybrid	Gersh	0.10	14.6 %	0.09	0.11	13.2 %	15.9 %
Hybrid	I-TSCA	0.19	24.1 %	0.19	0.20	22.8 %	25.3 %
Hybrid	Fixed	0.53	46.3 %	0.52	0.54	45.4 %	47.2 %
Gersh	I-TSCA	0.09	11.1 %	0.08	0.10	9.9 %	12.4 %
I-TSCA	Fixed	0.33	29.2 %	0.32	0.34	28.4 %	30.1 %

Table B.5: *Algorithmic comparison and ranking with respect to the mean normalised number of stops made by vehicles for the scenario in which $\lambda = 10$ for the corridor network topology.*

Traffic control algorithm 1	2	Average Improvement (s)	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
Gersh	Hybrid	0.66	3.0 %	0.20	1.12	0.9 %	5.2 %
Gersh	I-TSCA	3.43	14.0 %	2.97	3.89	12.2 %	15.9 %
Gersh	O-TSCA	3.46	14.1 %	3.00	3.92	12.3 %	16.0 %
Gersh	Fixed	4.65	18.1 %	4.19	5.12	16.3 %	19.9 %
Gersh	LH	7.61	26.6 %	7.14	8.07	25.0 %	28.2 %
Hybrid	I-TSCA	2.77	11.3 %	2.31	3.23	9.5 %	13.2 %
Hybrid	O-TSCA	2.80	11.4 %	2.34	3.26	9.6 %	13.3 %
Hybrid	Fixed	4.00	15.6 %	3.53	4.46	13.8 %	17.4 %
Hybrid	LH	6.95	24.3 %	6.49	7.41	22.7 %	25.9 %
I-TSCA	O-TSCA	0.03	0.1 %	−0.43	0.49	−1.8 %	2.0 %
I-TSCA	Fixed	1.22	4.8 %	0.76	1.69	3.0 %	6.6 %
I-TSCA	LH	4.18	14.6 %	3.72	4.64	13.0 %	16.2 %
O-TSCA	Fixed	1.20	4.7 %	0.74	1.66	2.9 %	6.5 %
Fixed	LH	2.95	10.3 %	2.49	3.41	8.7 %	11.9 %

Table B.6: *Algorithmic comparison and ranking with respect to the mean delay time for the scenario in which* $\lambda = 20$ *for the corridor network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
Gersh	Hybrid	0.02	1.4 %	0.01	0.02	1.0 %	1.8 %
Gersh	Fixed	0.05	3.5 %	0.04	0.05	3.1 %	3.9 %
Gersh	I-TSCA	0.07	4.9 %	0.06	0.07	4.5 %	5.3 %
Gersh	LH	0.08	5.6 %	0.07	0.08	5.2 %	6.0 %
Gersh	O-TSCA	0.08	5.9 %	0.08	0.09	5.5 %	6.3 %
Hybrid	Fixed	0.03	2.1 %	0.02	0.03	1.7 %	2.5 %
Hybrid	I-TSCA	0.05	3.6 %	0.04	0.06	3.2 %	4.0 %
Hybrid	LH	0.06	4.3 %	0.05	0.07	3.9 %	4.7 %
Hybrid	O-TSCA	0.06	4.6 %	0.06	0.07	4.2 %	5.0 %
Fixed	I-TSCA	0.02	1.5 %	0.02	0.03	1.1 %	1.9 %
Fixed	LH	0.03	2.2 %	0.03	0.04	1.8 %	2.6 %
Fixed	O-TSCA	0.04	2.5 %	0.03	0.04	2.1 %	2.9 %
I-TSCA	LH	0.01	0.7 %	0.00	0.02	0.3 %	1.1 %
LH	O-TSCA	0.00	0.3 %	0.00	0.01	−0.1 %	0.7 %

Table B.7: *Algorithmic comparison and ranking with respect to the mean normalised delay time for the scenario in which* $\lambda = 20$ *for the corridor network topology.*

Traffic control algorithm 1	2	Average Improvement (s)	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
Hybrid	O-TSCA	1.77	1.4 %	−6.59	10.12	−5.4 %	8.3 %
Hybrid	Gersh	6.69	5.3 %	−1.66	15.04	−1.3 %	11.8 %
Hybrid	I-TSCA	10.76	8.2 %	2.41	19.11	1.8 %	14.6 %
Hybrid	Fixed	25.25	17.3 %	16.90	33.60	11.6 %	23.1 %
Hybrid	LH	82.72	40.7 %	74.37	91.07	36.6 %	44.9 %
O-TSCA	Gersh	4.93	3.9 %	−3.42	13.28	−2.7 %	10.5 %
O-TSCA	I-TSCA	8.99	6.9 %	0.64	17.34	0.5 %	13.2 %
O-TSCA	Fixed	23.48	16.1 %	15.13	31.83	10.4 %	21.9 %
O-TSCA	LH	80.95	39.9 %	72.60	89.30	35.8 %	44.0 %
Gersh	I-TSCA	4.07	3.1 %	−4.29	12.42	−3.3 %	9.5 %
Gersh	Fixed	18.55	12.8 %	10.20	26.91	7.0 %	18.5 %
Gersh	LH	76.03	37.5 %	67.67	84.38	33.3 %	41.6 %
I-TSCA	Fixed	14.49	10.0 %	6.14	22.84	4.2 %	15.7 %
Fixed	LH	57.47	28.3 %	49.12	65.82	24.2 %	32.4 %

Table B.8: *Algorithmic comparison and ranking with respect to the maximum delay time for the scenario in which $\lambda = 20$ for the corridor network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
O-TSCA	Hybrid	0.10	8.7 %	0.08	0.12	6.9 %	10.6 %
O-TSCA	LH	0.30	22.9 %	0.28	0.32	21.3 %	24.4 %
O-TSCA	Gersh	0.32	24.2 %	0.30	0.34	22.6 %	25.7 %
O-TSCA	I-TSCA	0.36	26.4 %	0.34	0.38	24.9 %	27.9 %
O-TSCA	Fixed	0.63	38.6 %	0.61	0.65	37.3 %	39.8 %
Hybrid	LH	0.20	15.5 %	0.18	0.22	13.9 %	17.0 %
Hybrid	Gersh	0.22	16.9 %	0.20	0.24	15.4 %	18.4 %
Hybrid	I-TSCA	0.26	19.4 %	0.24	0.28	17.9 %	20.8 %
Hybrid	Fixed	0.53	32.7 %	0.51	0.55	31.4 %	33.9 %
LH	Gersh	0.02	1.7 %	0.00	0.04	0.1 %	3.2 %
LH	I-TSCA	0.06	4.6 %	0.04	0.08	3.1 %	6.1 %
LH	Fixed	0.33	20.3 %	0.31	0.35	19.1 %	21.6 %
Gersh	I-TSCA	0.04	3.0 %	0.02	0.06	1.5 %	4.4 %
I-TSCA	Fixed	0.27	16.5 %	0.25	0.29	15.3 %	17.7 %

Table B.9: *Algorithmic comparison and ranking with respect to the mean number of stops made by vehicles for the scenario in which $\lambda = 20$ for the corridor network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
O-TSCA	Hybrid	0.00	0.4 %	−0.01	0.01	−1.0 %	1.8 %
O-TSCA	LH	0.05	6.3 %	0.04	0.06	5.0 %	7.6 %
O-TSCA	Gersh	0.12	14.6 %	0.11	0.13	13.4 %	15.8 %
O-TSCA	I-TSCA	0.20	22.6 %	0.19	0.21	21.5 %	23.7 %
O-TSCA	Fixed	0.26	27.4 %	0.25	0.27	26.4 %	28.5 %
Hybrid	LH	0.04	5.9 %	0.03	0.05	4.6 %	7.3 %
Hybrid	Gersh	0.11	14.3 %	0.11	0.12	13.1 %	15.5 %
Hybrid	I-TSCA	0.20	22.3 %	0.19	0.21	21.2 %	23.4 %
Hybrid	Fixed	0.26	27.2 %	0.25	0.27	26.1 %	28.2 %
LH	Gersh	0.07	8.9 %	0.06	0.08	7.6 %	10.1 %
LH	I-TSCA	0.15	17.4 %	0.14	0.16	16.3 %	18.5 %
LH	Fixed	0.21	22.6 %	0.20	0.22	21.5 %	23.6 %
Gersh	I-TSCA	0.08	9.3 %	0.07	0.09	8.2 %	10.4 %
I-TSCA	Fixed	0.06	6.3 %	0.05	0.07	5.2 %	7.3 %

Table B.10: *Algorithmic comparison and ranking with respect to the mean normalised number of stops made by vehicles for the scenario in which $\lambda = 20$ for the corridor network topology.*

Traffic control algorithm 1	2	Average Improvement (s)	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
Fixed	Hybrid	1.49	4.3 %	−7.68	10.66	−22.3 %	31.0 %
Fixed	O-TSCA	3.36	9.3 %	−5.81	12.53	−16.0 %	34.5 %
Fixed	LH	7.16	17.9 %	−2.01	16.33	−5.0 %	40.8 %
Fixed	Gersh	16.70	33.7 %	7.53	25.87	15.2 %	52.2 %
Fixed	I-TSCA	28.87	46.7 %	19.70	38.04	31.9 %	61.6 %
Hybrid	O-TSCA	1.86	5.1 %	−7.31	11.03	−20.2 %	30.4 %
Hybrid	LH	5.66	14.1 %	−3.51	14.83	−8.8 %	37.0 %
Hybrid	Gersh	15.21	30.7 %	6.04	24.38	12.2 %	49.1 %
Hybrid	I-TSCA	27.38	44.3 %	18.21	36.55	29.5 %	59.2 %
O-TSCA	LH	3.80	9.5 %	−5.37	12.97	−13.4 %	32.4 %
O-TSCA	Gersh	13.35	26.9 %	4.18	22.52	8.4 %	45.4 %
O-TSCA	I-TSCA	25.52	41.3 %	16.35	34.69	26.5 %	56.1 %
LH	Gersh	9.55	19.2 %	0.38	18.72	0.8 %	37.7 %
Gersh	I-TSCA	12.17	19.7 %	3.00	21.34	4.9 %	34.5 %

Table B.11: *Algorithmic comparison and ranking with respect to the mean delay time for the scenario in which $\lambda = 30$ for the corridor network topology.*

Traffic control algorithm		Average Improvement		Interval limit		Interval limit	
1	2		(%)	Lower	Upper	Lower (%)	Upper (%)
Fixed	Hybrid	0.02	1.0 %	−0.13	0.16	−8.5 %	10.5 %
Fixed	O-TSCA	0.03	2.0 %	−0.12	0.18	−7.4 %	11.4 %
Fixed	LH	0.06	3.7 %	−0.09	0.21	−5.6 %	12.9 %
Fixed	Gersh	0.25	14.0 %	0.10	0.40	5.7 %	22.3 %
Fixed	I-TSCA	0.43	21.9 %	0.28	0.58	14.4 %	29.4 %
Hybrid	O-TSCA	0.02	1.0 %	−0.13	0.16	−8.4 %	10.4 %
Hybrid	LH	0.04	2.7 %	−0.10	0.19	−6.5 %	12.0 %
Hybrid	Gersh	0.23	13.1 %	0.09	0.38	4.9 %	21.4 %
Hybrid	I-TSCA	0.41	21.1 %	0.27	0.56	13.6 %	28.6 %
O-TSCA	LH	0.03	1.7 %	−0.12	0.17	−7.5 %	11.0 %
O-TSCA	Gersh	0.22	12.2 %	0.07	0.36	4.0 %	20.5 %
O-TSCA	I-TSCA	0.40	20.3 %	0.25	0.54	12.8 %	27.8 %
LH	Gersh	0.19	10.7 %	0.04	0.34	2.4 %	19.0 %
Gersh	I-TSCA	0.18	9.2 %	0.03	0.33	1.7 %	16.7 %

Table B.12: *Algorithmic comparison and ranking with respect to the mean normalised delay time for the scenario in which $\lambda = 30$ for the corridor network topology.*

Traffic control algorithm		Average Improvement		Interval limit		Interval limit	
1	2	(s)	(%)	Lower	Upper	Lower (%)	Upper (%)
Fixed	Hybrid	15.69	8.7 %	−96.37	127.75	−53.6 %	71.1 %
Fixed	O-TSCA	25.40	13.4 %	−86.66	137.46	−45.8 %	72.6 %
Fixed	LH	61.35	27.2 %	−50.71	173.41	−22.5 %	77.0 %
Fixed	Gersh	166.75	50.4 %	54.69	278.81	16.5 %	84.3 %
Fixed	I-TSCA	200.06	55.0 %	88.00	312.12	24.2 %	85.7 %
Hybrid	O-TSCA	9.71	5.1 %	−102.35	121.77	−54.0 %	64.3 %
Hybrid	LH	45.66	20.3 %	−66.40	157.72	−29.5 %	70.0 %
Hybrid	Gersh	151.06	45.7 %	39.00	263.12	11.8 %	79.6 %
Hybrid	I-TSCA	184.37	50.6 %	72.31	296.43	19.9 %	81.4 %
O-TSCA	LH	35.94	16.0 %	−76.12	148.00	−33.8 %	65.7 %
O-TSCA	Gersh	141.35	42.7 %	29.29	253.41	8.9 %	76.6 %
O-TSCA	I-TSCA	174.66	48.0 %	62.60	286.72	17.2 %	78.8 %
LH	Gersh	105.41	31.9 %	−6.65	217.47	-2.0 %	65.7 %
Gersh	I-TSCA	33.31	9.1 %	−78.75	145.37	−21.6 %	39.9 %

Table B.13: *Algorithmic comparison and ranking with respect to the maximum delay time for the scenario in which $\lambda = 30$ for the corridor network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
Fixed	Hybrid	0.03	2.2 %	−0.22	0.27	−16.9 %	21.2 %
Fixed	Gersh	0.05	3.6 %	−0.20	0.29	−15.1 %	22.3 %
Fixed	O-TSCA	0.14	10.4 %	−0.10	0.39	−7.1 %	27.8 %
Fixed	LH	0.34	21.5 %	0.10	0.58	6.2 %	36.7 %
Fixed	I-TSCA	1.10	46.8 %	0.86	1.34	36.5 %	57.2 %
Hybrid	Gersh	0.02	1.5 %	−0.22	0.26	−17.2 %	20.2 %
Hybrid	O-TSCA	0.12	8.4 %	−0.13	0.36	−9.0 %	25.8 %
Hybrid	LH	0.31	19.7 %	0.07	0.56	4.5 %	35.0 %
Hybrid	I-TSCA	1.07	45.7 %	0.83	1.32	35.4 %	56.0 %
Gersh	O-TSCA	0.10	7.0 %	−0.15	0.34	−10.4 %	24.4 %
Gersh	LH	0.29	18.5 %	0.05	0.54	3.3 %	33.8 %
Gersh	I-TSCA	1.05	44.8 %	0.81	1.30	34.5 %	55.2 %
O-TSCA	LH	0.20	12.4 %	−0.05	0.44	−2.9 %	27.6 %
LH	I-TSCA	0.76	32.3 %	0.52	1.00	22.0 %	42.6 %

Table B.14: *Algorithmic comparison and ranking with respect to the mean number of stops made by vehicles for the scenario in which $\lambda = 30$ for the corridor network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
Hybrid	Fixed	0.00	0.1 %	−0.15	0.16	−19.0 %	19.3 %
Hybrid	Gersh	0.01	0.7 %	−0.15	0.16	−18.3 %	19.8 %
Hybrid	O-TSCA	0.06	7.1 %	−0.09	0.22	−10.7 %	24.9 %
Hybrid	LH	0.11	11.6 %	−0.05	0.26	−5.4 %	28.5 %
Hybrid	I-TSCA	0.67	45.5 %	0.52	0.83	35.1 %	56.0 %
Fixed	Gersh	0.01	0.6 %	−0.15	0.16	−18.4 %	19.7 %
Fixed	O-TSCA	0.06	7.0 %	−0.09	0.21	−10.8 %	24.8 %
Fixed	LH	0.10	11.5 %	−0.05	0.26	−5.5 %	28.4 %
Fixed	I-TSCA	0.67	45.5 %	0.52	0.83	35.0 %	55.9 %
Gersh	O-TSCA	0.06	6.4 %	−0.10	0.21	−11.4 %	24.2 %
Gersh	LH	0.10	10.9 %	−0.05	0.25	−6.0 %	27.9 %
Gersh	I-TSCA	0.67	45.1 %	0.51	0.82	34.7 %	55.6 %
O-TSCA	LH	0.04	4.8 %	−0.11	0.20	−12.1 %	21.8 %
LH	I-TSCA	0.57	38.4 %	0.41	0.72	27.9 %	48.8 %

Table B.15: *Algorithmic comparison and ranking with respect to the mean normalised number of stops made by vehicles for the scenario in which $\lambda = 30$ for the corridor network topology.*

B.2 Algorithm rankings for the grid network topology

The results contained in the tables presented in this section correspond to the results shown in Figures 7.12, 7.13 and 7.14 for low, medium and high average vehicle arrival rates to the grid network, respectively.

Traffic control algorithm		Average Improvement		Interval limit		Interval limit	
1	2	(s)	(%)	Lower	Upper	Lower (%)	Upper (%)
Hybrid	Gersh	0.98	3.6 %	0.69	1.26	2.5 %	4.6 %
Hybrid	O-TSCA	1.97	7.0 %	1.68	2.25	5.9 %	8.0 %
Hybrid	I-TSCA	3.62	12.1 %	3.33	3.90	11.1 %	13.0 %
Hybrid	Fixed	6.96	20.9 %	6.67	7.24	20.1 %	21.8 %
Hybrid	LH	10.63	28.8 %	10.35	10.92	28.0 %	29.6 %
Gersh	O-TSCA	0.99	3.5 %	0.70	1.27	2.5 %	4.5 %
Gersh	I-TSCA	2.64	8.8 %	2.35	2.92	7.9 %	9.8 %
Gersh	Fixed	5.98	18.0 %	5.69	6.26	17.1 %	18.8 %
Gersh	LH	9.66	26.1 %	9.37	9.94	25.4 %	26.9 %
O-TSCA	I-TSCA	1.65	5.5 %	1.36	1.94	4.6 %	6.5 %
O-TSCA	Fixed	4.99	15.0 %	4.70	5.28	14.1 %	15.9 %
O-TSCA	LH	8.67	23.5 %	8.38	8.95	22.7 %	24.2 %
I-TSCA	Fixed	3.34	10.0 %	3.05	3.62	9.2 %	10.9 %
Fixed	LH	3.68	10.0 %	3.39	3.96	9.2 %	10.7 %

Table B.16: *Algorithmic comparison and ranking with respect to the mean delay time for the scenario in which* $\lambda = 10$ *for the grid network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
Hybrid	Gersh	0.01	0.7 %	0.01	0.01	0.5 %	1.0 %
Hybrid	O-TSCA	0.02	1.7 %	0.02	0.02	1.5 %	1.9 %
Hybrid	I-TSCA	0.03	2.5 %	0.03	0.04	2.3 %	2.7 %
Hybrid	Fixed	0.07	5.1 %	0.06	0.07	4.9 %	5.3 %
Hybrid	LH	0.10	7.1 %	0.09	0.10	6.9 %	7.3 %
Gersh	O-TSCA	0.01	1.0 %	0.01	0.02	0.8 %	1.2 %
Gersh	I-TSCA	0.02	1.8 %	0.02	0.03	1.6 %	2.0 %
Gersh	Fixed	0.06	4.4 %	0.06	0.06	4.2 %	4.6 %
Gersh	LH	0.09	6.4 %	0.08	0.09	6.2 %	6.6 %
O-TSCA	I-TSCA	0.01	0.8 %	0.01	0.01	0.6 %	1.0 %
O-TSCA	Fixed	0.05	3.4 %	0.04	0.05	3.2 %	3.6 %
O-TSCA	LH	0.07	5.5 %	0.07	0.08	5.3 %	5.7 %
I-TSCA	Fixed	0.03	2.6 %	0.03	0.04	2.4 %	2.8 %
Fixed	LH	0.03	2.2 %	0.03	0.03	2.0 %	2.4 %

Table B.17: *Algorithmic comparison and ranking with respect to the mean normalised delay time for the scenario in which $\lambda = 10$ for the grid network topology.*

Traffic control algorithm 1	2	Average Improvement (s)	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
Gersh	Hybrid	3.19	3.2 %	−2.95	9.32	−3.0 %	9.5 %
Gersh	Fixed	14.35	13.1 %	8.22	20.49	7.5 %	18.7 %
Gersh	O-TSCA	18.18	16.0 %	12.04	24.31	10.6 %	21.4 %
Gersh	I-TSCA	21.41	18.3 %	15.28	27.55	13.1 %	23.6 %
Gersh	LH	53.78	36.1 %	47.65	59.92	32.0 %	40.2 %
Hybrid	Fixed	11.16	10.2 %	5.03	17.30	4.6 %	15.8 %
Hybrid	O-TSCA	14.99	13.2 %	8.86	21.12	7.8 %	18.6 %
Hybrid	I-TSCA	18.22	15.6 %	12.09	24.36	10.4 %	20.9 %
Hybrid	LH	50.60	33.9 %	44.46	56.73	29.8 %	38.0 %
Fixed	O-TSCA	3.83	3.4 %	−2.31	9.96	−2.0 %	8.8 %
Fixed	I-TSCA	7.06	6.0 %	0.93	13.19	0.8 %	11.3 %
Fixed	LH	39.43	26.4 %	33.30	45.57	22.3 %	30.6 %
O-TSCA	I-TSCA	3.23	2.8 %	−2.90	9.37	−2.5 %	8.0 %
I-TSCA	LH	32.37	21.7 %	26.24	38.51	17.6 %	25.8 %

Table B.18: *Algorithmic comparison and ranking with respect to the maximum delay time for the scenario in which $\lambda = 10$ for the grid network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
O-TSCA	Hybrid	0.66	35.9 %	0.64	0.68	34.8 %	37.0 %
O-TSCA	Gersh	1.07	47.6 %	1.05	1.09	46.7 %	48.6 %
O-TSCA	I-TSCA	1.17	49.9 %	1.15	1.19	49.0 %	50.7 %
O-TSCA	LH	1.61	57.8 %	1.59	1.63	57.0 %	58.5 %
O-TSCA	Fixed	2.09	64.1 %	2.07	2.11	63.4 %	64.7 %
Hybrid	Gersh	0.41	18.3 %	0.39	0.43	17.4 %	19.2 %
Hybrid	I-TSCA	0.51	21.8 %	0.49	0.53	20.9 %	22.7 %
Hybrid	LH	0.95	34.1 %	0.93	0.97	33.4 %	34.9 %
Hybrid	Fixed	1.43	43.9 %	1.41	1.46	43.3 %	44.6 %
Gersh	I-TSCA	0.10	4.3 %	0.08	0.12	3.4 %	5.1 %
Gersh	LH	0.54	19.4 %	0.52	0.56	18.6 %	20.1 %
Gersh	Fixed	1.02	31.4 %	1.00	1.04	30.7 %	32.0 %
I-TSCA	LH	0.44	15.8 %	0.42	0.46	15.0 %	16.5 %
LH	Fixed	0.49	14.9 %	0.47	0.51	14.2 %	15.5 %

Table B.19: *Algorithmic comparison and ranking with respect to the mean number of stops made by vehicles for the scenario in which $\lambda = 10$ for the grid network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
O-TSCA	Hybrid	0.14	42.4 %	0.13	0.15	40.5 %	44.3 %
O-TSCA	Gersh	0.24	56.4 %	0.24	0.25	54.9 %	57.8 %
O-TSCA	I-TSCA	0.28	59.7 %	0.27	0.29	58.4 %	61.1 %
O-TSCA	LH	0.40	68.0 %	0.40	0.41	66.9 %	69.1 %
O-TSCA	Fixed	0.55	74.4 %	0.54	0.55	73.5 %	75.2 %
Hybrid	Gersh	0.10	24.2 %	0.10	0.11	22.8 %	25.7 %
Hybrid	I-TSCA	0.14	30.1 %	0.13	0.15	28.8 %	31.4 %
Hybrid	LH	0.26	44.5 %	0.26	0.27	43.4 %	45.5 %
Hybrid	Fixed	0.41	55.5 %	0.40	0.42	54.6 %	56.3 %
Gersh	I-TSCA	0.04	7.7 %	0.03	0.04	6.4 %	9.1 %
Gersh	LH	0.16	26.7 %	0.15	0.16	25.6 %	27.8 %
Gersh	Fixed	0.30	41.3 %	0.30	0.31	40.4 %	42.1 %
I-TSCA	LH	0.12	20.5 %	0.12	0.13	19.5 %	21.6 %
LH	Fixed	0.15	19.9 %	0.14	0.15	19.0 %	20.7 %

Table B.20: *Algorithmic comparison and ranking with respect to the mean normalised number of stops made by vehicles for the scenario in which $\lambda = 10$ for the grid network topology.*

Traffic control algorithm		Average Improvement		Interval limit		Interval limit	
1	2	(s)	(%)	Lower	Upper	Lower (%)	Upper (%)
O-TSCA	Gersh	6.32	15.1 %	5.86	6.79	14.0 %	16.2 %
O-TSCA	Hybrid	10.10	22.1 %	9.63	10.56	21.1 %	23.2 %
O-TSCA	I-TSCA	13.56	27.6 %	13.09	14.02	26.7 %	28.6 %
O-TSCA	Fixed	21.98	38.2 %	21.52	22.45	37.4 %	39.0 %
O-TSCA	LH	35.52	50.0 %	35.06	35.99	49.3 %	50.7 %
Gersh	Hybrid	3.78	8.3 %	3.31	4.24	7.3 %	9.3 %
Gersh	I-TSCA	7.24	14.7 %	6.77	7.70	13.8 %	15.7 %
Gersh	Fixed	15.66	27.2 %	15.20	16.13	26.4 %	28.0 %
Gersh	LH	29.20	41.1 %	28.74	29.67	40.4 %	41.8 %
Hybrid	I-TSCA	3.46	7.0 %	2.99	3.93	6.1 %	8.0 %
Hybrid	Fixed	11.89	20.7 %	11.42	12.35	19.9 %	21.5 %
Hybrid	LH	25.43	35.8 %	24.96	25.89	35.1 %	36.4 %
I-TSCA	Fixed	8.43	14.7 %	7.96	8.89	13.8 %	15.5 %
Fixed	LH	13.54	19.1 %	13.07	14.01	18.4 %	19.7 %

Table B.21: *Algorithmic comparison and ranking with respect to the mean delay time for the scenario in which $\lambda = 20$ for the grid network topology.*

Traffic control algorithm		Average Improvement		Interval limit		Interval limit	
1	2		(%)	Lower	Upper	Lower (%)	Upper (%)
O-TSCA	Gersh	0.06	3.9 %	0.05	0.06	3.6 %	4.2 %
O-TSCA	Hybrid	0.09	6.4 %	0.09	0.10	6.1 %	6.7 %
O-TSCA	I-TSCA	0.12	7.9 %	0.11	0.12	7.7 %	8.2 %
O-TSCA	Fixed	0.20	12.8 %	0.19	0.20	12.5 %	13.1 %
O-TSCA	LH	0.32	19.3 %	0.32	0.33	19.0 %	19.5 %
Gersh	Hybrid	0.04	2.6 %	0.03	0.04	2.3 %	2.9 %
Gersh	I-TSCA	0.06	4.2 %	0.06	0.07	3.9 %	4.5 %
Gersh	Fixed	0.14	9.2 %	0.14	0.15	9.0 %	9.5 %
Gersh	LH	0.27	16.0 %	0.26	0.27	15.7 %	16.2 %
Hybrid	I-TSCA	0.02	1.6 %	0.02	0.03	1.4 %	1.9 %
Hybrid	Fixed	0.11	6.8 %	0.10	0.11	6.6 %	7.1 %
Hybrid	LH	0.23	13.8 %	0.23	0.23	13.5 %	14.0 %
I-TSCA	Fixed	0.08	5.3 %	0.08	0.09	5.0 %	5.5 %
Fixed	LH	0.12	7.4 %	0.12	0.13	7.2 %	7.7 %

Table B.22: *Algorithmic comparison and ranking with respect to the mean normalised delay time for the scenario in which $\lambda = 20$ for the grid network topology.*

Traffic control algorithm 1	2	Average Improvement (s)	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
O-TSCA	Gersh	19.55	15.2 %	8.27	30.84	6.4 %	24.0 %
O-TSCA	Hybrid	30.97	22.1 %	19.68	42.25	14.1 %	30.2 %
O-TSCA	Fixed	31.86	22.6 %	20.57	43.15	14.6 %	30.6 %
O-TSCA	I-TSCA	58.88	35.1 %	47.59	70.17	28.4 %	41.8 %
O-TSCA	LH	144.31	57.0 %	133.02	155.60	52.5 %	61.4 %
Gersh	Hybrid	11.41	8.2 %	0.12	22.70	0.1 %	16.2 %
Gersh	Fixed	12.30	8.7 %	1.02	23.59	0.7 %	16.8 %
Gersh	I-TSCA	39.33	23.4 %	28.04	50.61	16.7 %	30.2 %
Gersh	LH	124.75	49.3 %	113.47	136.04	44.8 %	53.7 %
Hybrid	Fixed	0.89	0.6 %	−10.39	12.18	−7.4 %	8.7 %
Hybrid	I-TSCA	27.91	16.6 %	16.63	39.20	9.9 %	23.4 %
Hybrid	LH	113.34	44.8 %	102.06	124.63	40.3 %	49.2 %
Fixed	I-TSCA	27.02	16.1 %	15.73	38.31	9.4 %	22.8 %
I-TSCA	LH	85.43	33.7 %	74.14	96.72	29.3 %	38.2 %

Table B.23: *Algorithmic comparison and ranking with respect to the maximum delay time for the scenario in which $\lambda = 20$ for the grid network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
O-TSCA	Hybrid	0.67	34.2 %	0.65	0.69	33.0 %	35.4 %
O-TSCA	Gersh	1.19	48.1 %	1.17	1.21	47.2 %	49.0 %
O-TSCA	LH	1.47	53.4 %	1.45	1.49	52.5 %	54.2 %
O-TSCA	I-TSCA	1.55	54.7 %	1.53	1.57	53.9 %	55.5 %
O-TSCA	Fixed	1.87	59.3 %	1.85	1.89	58.5 %	60.0 %
Hybrid	Gersh	0.52	21.1 %	0.50	0.55	20.2 %	22.1 %
Hybrid	LH	0.80	29.1 %	0.78	0.83	28.3 %	30.0 %
Hybrid	I-TSCA	0.88	31.1 %	0.86	0.91	30.3 %	31.9 %
Hybrid	Fixed	1.20	38.1 %	1.18	1.23	37.4 %	38.8 %
Gersh	LH	0.28	10.1 %	0.26	0.30	9.3 %	11.0 %
Gersh	I-TSCA	0.36	12.7 %	0.34	0.38	11.8 %	13.5 %
Gersh	Fixed	0.68	21.5 %	0.66	0.70	20.8 %	22.2 %
LH	I-TSCA	0.08	2.8 %	0.06	0.10	2.0 %	3.6 %
I-TSCA	Fixed	0.32	10.1 %	0.30	0.34	9.4 %	10.9 %

Table B.24: *Algorithmic comparison and ranking with respect to the mean number of stops made by vehicles for the scenario in which $\lambda = 20$ for the grid network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
O-TSCA	Hybrid	0.15	41.5 %	0.14	0.15	39.5 %	43.4 %
O-TSCA	Gersh	0.29	58.5 %	0.29	0.30	57.2 %	59.9 %
O-TSCA	LH	0.36	63.2 %	0.35	0.37	62.0 %	64.4 %
O-TSCA	I-TSCA	0.40	65.8 %	0.40	0.41	64.7 %	67.0 %
O-TSCA	Fixed	0.49	70.0 %	0.48	0.49	69.0 %	71.0 %
Hybrid	Gersh	0.15	29.1 %	0.14	0.15	27.7 %	30.5 %
Hybrid	LH	0.21	37.1 %	0.20	0.22	35.9 %	38.4 %
Hybrid	I-TSCA	0.25	41.6 %	0.25	0.26	40.5 %	42.7 %
Hybrid	Fixed	0.34	48.8 %	0.33	0.35	47.8 %	49.7 %
Gersh	LH	0.06	11.3 %	0.06	0.07	10.1 %	12.5 %
Gersh	I-TSCA	0.11	17.6 %	0.10	0.11	16.5 %	18.8 %
Gersh	Fixed	0.19	27.7 %	0.19	0.20	26.7 %	28.7 %
LH	I-TSCA	0.04	7.1 %	0.04	0.05	6.0 %	8.2 %
I-TSCA	Fixed	0.09	12.2 %	0.08	0.09	11.2 %	13.2 %

Table B.25: *Algorithmic comparison and ranking with respect to the mean normalised number of stops made by vehicles for the scenario in which $\lambda = 20$ for the grid network topology.*

Traffic control algorithm 1	2	Average Improvement (s)	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
O-TSCA	Fixed	10.88	17.3 %	7.58	14.18	12.0 %	22.5 %
O-TSCA	Hybrid	12.77	19.7 %	9.46	16.07	14.6 %	24.7 %
O-TSCA	Gersh	20.50	28.2 %	17.19	23.80	23.7 %	32.7 %
O-TSCA	LH	36.22	41.0 %	32.92	39.52	37.2 %	44.7 %
O-TSCA	I-TSCA	40.28	43.6 %	36.98	43.58	40.0 %	47.1 %
Fixed	Hybrid	1.89	2.9 %	−1.41	5.19	−2.2 %	8.0 %
Fixed	Gersh	9.62	13.2 %	6.31	12.92	8.7 %	17.8 %
Fixed	LH	25.34	28.7 %	22.04	28.64	24.9 %	32.4 %
Fixed	I-TSCA	29.40	31.8 %	26.10	32.70	28.2 %	35.4 %
Hybrid	Gersh	7.73	10.6 %	4.43	11.03	6.1 %	15.2 %
Hybrid	LH	23.45	26.5 %	20.15	26.75	22.8 %	30.3 %
Hybrid	I-TSCA	27.51	29.8 %	24.21	30.82	26.2 %	33.3 %
Gersh	LH	15.72	17.8 %	12.42	19.03	14.0 %	21.5 %
LH	I-TSCA	4.06	4.4 %	0.76	7.36	0.8 %	8.0 %

Table B.26: *Algorithmic comparison and ranking with respect to the mean delay time for the scenario in which $\lambda = 30$ for the grid network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
O-TSCA	Fixed	0.10	6.2 %	0.07	0.13	4.3 %	8.1 %
O-TSCA	Hybrid	0.12	7.3 %	0.09	0.15	5.5 %	9.2 %
O-TSCA	Gersh	0.19	11.2 %	0.16	0.22	9.4 %	13.0 %
O-TSCA	LH	0.32	17.5 %	0.29	0.35	15.9 %	19.2 %
O-TSCA	I-TSCA	0.37	19.8 %	0.34	0.40	18.2 %	21.4 %
Fixed	Hybrid	0.02	1.2 %	−0.01	0.05	−0.7 %	3.1 %
Fixed	Gersh	0.09	5.3 %	0.06	0.12	3.5 %	7.1 %
Fixed	LH	0.22	12.1 %	0.19	0.25	10.4 %	13.7 %
Fixed	I-TSCA	0.27	14.5 %	0.24	0.30	12.8 %	16.1 %
Hybrid	Gersh	0.07	4.1 %	0.04	0.10	2.3 %	5.9 %
Hybrid	LH	0.20	11.0 %	0.17	0.23	9.3 %	12.7 %
Hybrid	I-TSCA	0.25	13.4 %	0.22	0.28	11.8 %	15.1 %
Gersh	LH	0.13	7.2 %	0.10	0.16	5.5 %	8.9 %
LH	I-TSCA	0.05	2.7 %	0.02	0.08	1.1 %	4.4 %

Table B.27: *Algorithmic comparison and ranking with respect to the mean normalised delay time for the scenario in which $\lambda = 30$ for the grid network topology.*

Traffic control algorithm 1	2	Average Improvement (s)	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
Fixed	O-TSCA	15.29	8.9 %	−8.82	39.39	−5.2 %	23.0 %
Fixed	Hybrid	46.42	23.0 %	22.31	70.53	11.0 %	34.9 %
Fixed	Gersh	81.72	34.4 %	57.61	105.83	24.3 %	44.6 %
Fixed	I-TSCA	88.10	36.1 %	63.99	112.20	26.2 %	46.0 %
Fixed	LH	127.91	45.1 %	103.80	152.01	36.6 %	53.6 %
O-TSCA	Hybrid	31.13	15.4 %	7.03	55.24	3.5 %	27.3 %
O-TSCA	Gersh	66.43	28.0 %	42.33	90.54	17.8 %	38.1 %
O-TSCA	I-TSCA	72.81	29.9 %	48.71	96.92	20.0 %	39.7 %
O-TSCA	LH	112.62	39.7 %	88.51	136.73	31.2 %	48.2 %
Hybrid	Gersh	35.30	14.9 %	11.19	59.41	4.7 %	25.0 %
Hybrid	I-TSCA	41.68	17.1 %	17.57	65.79	7.2 %	27.0 %
Hybrid	LH	81.49	28.7 %	57.38	105.59	20.2 %	37.2 %
Gersh	I-TSCA	6.38	2.6 %	−17.73	30.48	−7.3 %	12.5 %
I-TSCA	LH	39.81	14.0 %	15.70	63.91	5.5 %	22.5 %

Table B.28: *Algorithmic comparison and ranking with respect to the maximum delay time for the scenario in which $\lambda = 30$ for the grid network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
O-TSCA	Hybrid	0.28	12.6 %	0.21	0.34	9.4 %	15.8 %
O-TSCA	Fixed	0.42	17.9 %	0.35	0.48	14.9 %	20.9 %
O-TSCA	Gersh	0.48	20.2 %	0.41	0.55	17.3 %	23.1 %
O-TSCA	LH	1.22	39.0 %	1.15	1.29	36.8 %	41.2 %
O-TSCA	I-TSCA	1.76	47.9 %	1.69	1.83	46.0 %	49.8 %
Hybrid	Fixed	0.14	6.0 %	0.07	0.21	3.0 %	9.0 %
Hybrid	Gersh	0.21	8.7 %	0.14	0.28	5.8 %	11.6 %
Hybrid	LH	0.95	30.2 %	0.88	1.02	28.0 %	32.4 %
Hybrid	I-TSCA	1.48	40.4 %	1.41	1.55	38.5 %	42.3 %
Fixed	Gersh	0.07	2.9 %	0.00	0.14	0.0 %	5.8 %
Fixed	LH	0.81	25.7 %	0.74	0.88	23.5 %	28.0 %
Fixed	I-TSCA	1.34	36.6 %	1.27	1.41	34.7 %	38.5 %
Gersh	LH	0.74	23.5 %	0.67	0.81	21.3 %	25.8 %
LH	I-TSCA	0.54	14.6 %	0.47	0.61	12.7 %	16.5 %

Table B.29: *Algorithmic comparison and ranking with respect to the mean number of stops made by vehicles for the scenario in which $\lambda = 30$ for the grid network topology.*

Traffic control algorithm 1	2	Average Improvement	(%)	Interval limit Lower	Upper	Interval limit Lower (%)	Upper (%)
O-TSCA	Hybrid	0.07	17.7 %	0.05	0.09	12.9 %	22.6 %
O-TSCA	Fixed	0.10	22.5 %	0.08	0.12	18.0 %	27.1 %
O-TSCA	Gersh	0.13	27.6 %	0.11	0.15	23.3 %	31.8 %
O-TSCA	LH	0.32	48.4 %	0.30	0.34	45.4 %	51.5 %
O-TSCA	I-TSCA	0.53	60.6 %	0.51	0.55	58.3 %	62.9 %
Hybrid	Fixed	0.03	5.8 %	0.01	0.05	1.3 %	10.4 %
Hybrid	Gersh	0.06	12.0 %	0.04	0.08	7.7 %	16.2 %
Hybrid	LH	0.25	37.3 %	0.23	0.27	34.3 %	40.3 %
Hybrid	I-TSCA	0.46	52.1 %	0.43	0.48	49.8 %	54.4 %
Fixed	Gersh	0.03	6.5 %	0.01	0.05	2.2 %	10.8 %
Fixed	LH	0.22	33.4 %	0.20	0.24	30.4 %	36.5 %
Fixed	I-TSCA	0.43	49.2 %	0.41	0.45	46.8 %	51.5 %
Gersh	LH	0.19	28.8 %	0.17	0.21	25.8 %	31.8 %
LH	I-TSCA	0.21	23.6 %	0.19	0.23	21.3 %	26.0 %

Table B.30: *Algorithmic comparison and ranking with respect to the mean normalised number of stops made by vehicles for the scenario in which $\lambda = 30$ for the grid network topology.*

APPENDIX C

Contents of the accompanying compact disk

This appendix contains a brief description of the contents of the compact disc included with this dissertation. The compact disc contains an electronic version of the dissertation itself in ".pdf" format, as well as three example videos in ".mp4" format illustrating the performances of the I-TSCA, the O-TSCA and the Hybrid traffic control algorithm within the microscopic traffic simulation modelling framework described in Chapter 4. The compact disc contains the following two directories:

Dissertation. This directory contains an electronic copy of the dissertation in ".pdf" format.

Example videos. This directory contains three example videos. The video entitled *I-TSCA.mp4* is an example of the functioning of the I-TSCA. In this video, the chart on the left depicts the evolution of the cost in terms of vehicle delay time for providing service to vehicles travelling from west to east or from east to west (purple line) as well as the cost for providing service to vehicles travelling from north to south or from south to north (blue line). The chart on the right depicts the remaining green time of each phase, as well as the average green times received by each phase. The purple lines, once again, correspond to the phase during which vehicles travelling from west to east or east to west receive service, while the blue lines correspond to the phase during which vehicles travelling from north to south or from south to north receive service. The slider labelled *West to East Arrival Rate* may be used to control the average rate of arrival of all vehicles travelling from west to east. After thirty-five seconds, the average rate of arrival of vehicles travelling from west to east is increased from 5 vehicles per minute to 20 vehicles per minute and the speed of the simulation model is increased to illustrate how the algorithm automatically adjusts to the increase in vehicle arrivals by lengthening the green time durations, as illustrated by the increase in average green time values.

The video entitled *O-TSCA.mp4* demonstrates the functioning of the O-TSCA. In the video, two adjacent intersections may be observed. The O-TSCA is implemented at the intersection on the left, while a simple-fixed time algorithm is implemented at the intersection on the right. This is done to allow saturation to occur so as to demonstrate the ability of the O-TSCA to prevent queue spill-backs under heavy traffic flow conditions. The chart on the left depicts the pressures exerted on the intersection by the two competing phases. The purple line, once again, corresponds to the phase during which vehicles travelling from west to east or from east to west receive service, while the blue line corresponds to the phase during which vehicles travelling from north to south or from south to north receive service. The chart on the right depicts the throughput (black line), the demand (green line) and the availability (red line) of the phase during which vehicles travelling from west

to east or from east to west are served. Under the initial light traffic flow conditions (*i.e.* when the average vehicle arrival rate of vehicles travelling from west to east was set to 5 vehicles per minute) it may be observed from the chart on the right that the availability exceeds the demand and that the throughput never exceeds this demand. After 27 seconds, however, the video is fast-forwarded to more saturated conditions for which the average vehicle arrival rate is 20 vehicles per minute. Under these conditions, it may be seen from the chart on the right that the demand exceeds the availability. A signal switch is, however, implemented as soon as the throughput equals the availability, thus preventing spill-backs.

The video entitled *Hybrid.mp4* demonstrates the functioning of the Hybrid traffic control algorithm. Initially, the average vehicle arrival rate of vehicles travelling from west to east is set to twenty vehicles per minute. This is done to demonstrate that under heavy traffic flow conditions, Hybrid permits vehicles to continue to enter into the intersection even once the throughput has equalled the initial demand, as determined by O-TSCA and illustrated in the chart on the bottom right. After 30 seconds, the average vehicle arrival rate is reduced to 4.3 vehicles per minute. This demonstrates how Hybrid seamlessly begins to implement shorter, more flexible green times under light traffic signal conditions.