Wireless Vehicular Communication Based Solution for Road Traffic Efficiency

Kevin Collins

Bachelor of Engineering in Digital Media Engineering

A Dissertation submitted in fulfilment of the

requirements for the award of

Doctor of Philosophy (Ph.D.)

to the

Dublin City University

Faculty of Engineering and Computing, School of Engineering

Supervisor: Dr. Gabriel Miro Muntean

Ph.D. Defended December 2009

Final Submission January 2011

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed

Student ID: 52434547

Date

Acknowledgements

I would like to thank the Irish Research Council for Science, Engineering and Technology for granting me a scholarship through their EMBARK Initiative. Without their support this work would not have been possible. Once the financial support is in place a PhD student needs a good supervisor and I was very lucky with mine. I'd like to thank my supervisor Dr. Gabriel Miro Muntean co-Director of the Performance Engineering Laboratory at Dublin City University. Known as "Gabi" to friends and colleagues I am glad to say it was my pleasure to work with him for the duration of my PhD. Blessed with infectious enthusiasm and seemingly boundless energy Gabi was always generous with his time and a tireless advocate for me and all his students, so again Gabi, many thanks. I was also fortunate to have many wonderful colleagues during my time in the Performance Engineering Laboratory. The lab was a great place to work and their was always someone to bounce an idea off, proof read a paper, or look over your shoulder at a particularly pesky problem. I'd like to thank all my colleagues for their help during my time in the lab but some deserve a special mention. I started my Phd at the same time as a Korean gentleman by the name of Seung Bum Lee and we had our Phd Vivas about a week apart. It was great to have someone else in the same boat over the course of my PhD and Seung Bum was extremely helpful and a great support throughout. So to my good friend Dr. Lee I would like to say thanks for everything. Not too long after I started we were joined by two new comers who also became good friends, and whose help during my PhD I greatly appreciate - Sabine and Bogdan many thanks to you also. Finally I would like to thank my Mom and Dad for everything they have done for me over the years. A son could not ask for better parents and their unwavering support of my education and in all things has perhaps more than anything made this PhD possible.

Abstract

Wireless vehicular communications is a cutting edge set of technologies driven by the vision of providing a suite of original applications, and supported by emerging standards such as IEEE 802.11p. In turn the popularity of these applications is one of the key factors, which will drive the uptake of these vehicular communications technologies and ultimately determine their market success. Applications for vehicular communications can be placed in three main categories - Traffic Safety, Traffic Efficiency and Value-added Services (e.g. Infotainment/Business). Our work focuses on the provision of *traffic efficiency services* as we believe they offer an immediate benefit and can be adopted quickly by a large number of potential users. Satellite navigation systems provide a ready made deployment platform for these types of services and have already proven popular (14.4 million portable satellite navigation systems sold in Western Europe in 2007[1]). There is also an existing trend toward complementing satellite navigation-related technology with local area wireless communications (by 2013 34% of all portable navigation devices will feature wireless cards [2]). Our emphasis is on an *infrastructure-based approach* as this allows early adopters of wireless enabled satellite navigation devices to receive useful services from day one, regardless of the penetration level of the technology. This thesis describes *Smart City, a novel framework*, which purposes the use of wireless communication to make city life greener and more efficient. A major contribution to this framework is the proposed intelligent traffic management module. A route management service, which is powered by *a best route selection algorithm*, is put forward as a prototypical traffic efficiency service for this module. *The novel aspect is that the algorithm minimizes journey times and traffic congestion as well as fuel consumption and emissions.* Testing has shown how the algorithm provides - shorter journey times, a reduction in fuel consumption and harmful emissions and also results in financial savings. We have proposed and implemented an infrastructure-based communication scheme that enables prioritization of services provided to vehicles.

[1] I. Skog and P. Handel, ”In-car positioning and navigation technologies a survey,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 10, pp. 4-21, March 2009.

[2] ABI Research, ”Connected navigation devices,” tech. rep., ABI Research, April 2008.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Smart City

A future where wireless access is ubiquitous in urban areas around the world is fast approaching. In this climate there is huge potential to make city life more efficient by harnessing the power of wireless communications. Here is a simple scenario - imagine a citizen of the city getting up for work in the morning opening the fridge to realise they are out of milk but not needing to worry because the fridge has sensed this, ordered new supplies and they find a fresh bottle on the doorstep. After breakfast they get in their car to go to work, the car advises them on the fastest route to take based on current traffic conditions and directs them to the nearest available car space when they arrive. On the way home they are reminded to pick up groceries and refuel for their business trip tomorrow. The car accesses a retail database and calculates a fast route home which incorporates a stop at a retail outlet that meets all their needs at an acceptable price and a stop at the crche to collect their child while also minimising the time wasted in traffic.

It is in this context that we present the *"Smart City"* concept as seen in figure 1.1. Actors participate in the daily life of a city e.g. a private individual, a car, a traffic light, a fridge. If actors are enabled for communication (wireless or otherwise) then they may participate in the smart city by collaborating or communicating with other actors or with a public or private system, available city-wide which provides an array of life or urban experience en-

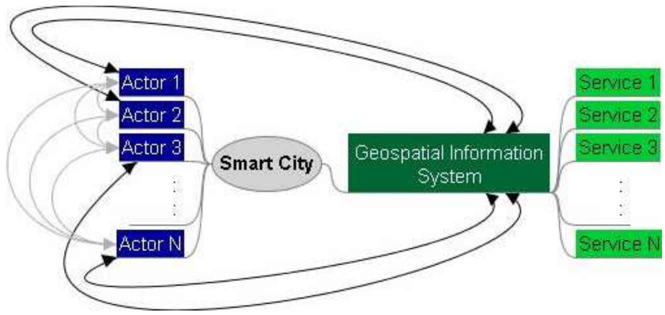


Figure 1.1: Smart City Concept

hancing services. The range of life/urban experience enhancing services is limited only by the number of actors capable of meaningful communication. What is meant by meaningful communication can be more clearly seen in the Smart City conceptual framework outlined in figure 1.2.

An actor has a communication manager which oversees two asynchronous modes of communication sensing and direct interfacing. In sensing mode the actor gathers and relays geospatially referenced information about itself and/or its surroundings to a geospatial information system. The information collected is determined by what is required to enable the services being provided. In direct interfacing mode the actor is availing of a service, i.e. some request is made and a suitable response is received. A practical scenario where vehicles are the main participant, interacting with a traffic management module is discussed next and is illustrated in figure 1.3.

Vehicles have the potential to play an important role within a Smart City for a variety of reasons. First off they are virtually ubiquitous in the urban environment, which makes them an excellent platform for a wireless device capable of Smart City participation. The same

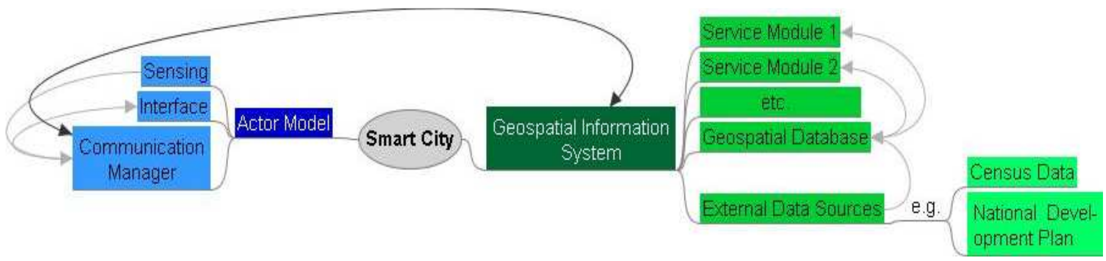


Figure 1.2: Smart City Conceptual Framework

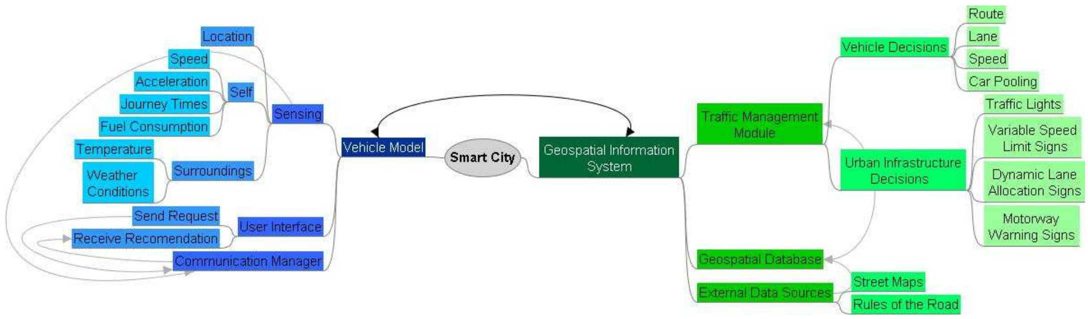


Figure 1.3: Traffic Management Module

could be said of people most of whom already carry a wireless device i.e. a mobile phone. However vehicles offer a number of distinct advantages - due to their size and constant power supply when in motion they do not have strict limits on -

- Processing power
- Storage capabilities
- Power consumption

These properties mean that wirelessly enabled vehicles can be a powerful application platform. The unique nature of the vehicular environment also invites the development of applications which are truly original.

To enable the traffic management service all vehicles in the city gather geospatially referenced data which they make available to the city geospatial information system. Some sample metrics for vehicles to measure are shown, i.e. speed, acceleration, journey time and fuel consumption. The geospatial information system stores the data in a database that the traffic management module can access. Some necessary external data sources which change less frequently are also made available, e.g. street maps and the rules of the road. The data will be used by the algorithms which power the services provided by the traffic management module.

One use case for the traffic management module goes as follows-

- A user enters a car to embark on a journey and begins by entering their destination and requesting directions via the on-board interface.

- Using up-to-date information on the state of the city supplied by other vehicles the traffic management module sends instructions to the user guiding them to their destination in the most efficient manner possible and even directing them to an open parking space.

There are other actors in the city aside from vehicles which could interact with the traffic module e.g. public infrastructure such as traffic lights, roadside signs, parking meters, pedestrians etc. However in order for vehicles to take part in the Smart City an emerging set of technologies known as Vehicular Communication must be used. In the next section Vehicular Communications and Traffic Efficiency issues are introduced and it is outlined how the former can be harnessed to support the latter in cities worldwide.

1.2 Vehicular Communications & Traffic Efficiency

The concept of wirelessly enabled vehicles is described as vehicular communications. Vehicular communication has garnered widespread attention in both industry and academia in recent times. The development of standards which enable it is well under way (e.g. IEEE1609 Wireless Access in Vehicular Environments (WAVE) [88, 49, 71, 72, 74, 70]). The biggest remaining obstacle for this technology is the emergence of a killer application to drive market introduction.

The services which are likely to be provided to vehicles via vehicular communications are typically classified as [97]:

- Traffic safety
- Traffic efficiency
- Value-added services (e.g., infotainment, business applications)

Numerous applications have been proposed for vehicular communications but safety applications have been most prominent, ongoing work in this area including [180, 182, 89]. The importance of road safety cannot be questioned however, many safety applications rely on Inter-vehicle Communications (IVC) and the formation of Vehicle Ad-hoc Networks

(VANET). Given the dearth of successfully deployed commercial examples of ad-hoc networks these types of applications are unlikely to be viable in the early stages of vehicular communications. Also certain safety applications by their very nature requires very high (up to 100%) penetration rates of the technology e.g. co-operative collision avoidance [151]. The distribution method is also a problem with safety features being either fitted as standard or provided as optional extras in new cars, after-market introduction seems unlikely. This reduces the potential for them to be introduced quickly.

The need for traffic efficiency solutions is clear, vehicular traffic is one of the most critical concerns for a modern society where cities are ever-growing. In 2008 for the first time, more than half of the world‘s population lives in urban areas and the balance of people continues to shift towards the cities [161]. It is well known that in urban areas commuters can spend a large percentage of their day stuck in traffic. It has been estimated that traffic congestion will cost the US economy over $90bn per year by 2009 [103] and the EU economy approximately 1% of its GDP by 2010 [47]. There is also the environmental cost. In Europe in 2004 road transport accounted for 19.5% of greenhouse gas emissions [46]. Alarmingly there is a worsening trend as the growth in the number of vehicles on the road out-paces growth in road capacity worldwide and the construction of new roads is ultimately constrained by space.

On a more personal level the benefit of traffic efficiency solutions to individual users is easy to justify (see figure 1.4. The most precious commodity any of us possesses is our time. Even a long human life adds up to only about 650,000 hours [22]. Remove an average of 8 hours a day for sleep and that leaves approximately 430,000 waking hours. Now take a typical vehicle user who has a sizeable commute to work, uses their transport at the weekend for errands and leisure, resulting in an average of 3hrs per day in transit. That adds up to over 80,000 hours or approximately one fifth of their waking life spent behind the wheel. Who would not want to reduce that time, to spend a little less time ”getting there” and a little more time being there doing something useful, something fun or just nothing at all?

The route to introduction of traffic efficiency services is clear-cut; they offer an immediate benefit and can be adopted quickly by a large number of potential users. Satellite

Figure 1.4: No one likes being stuck in traffic![1]

navigation systems provide a ready made deployment platform for these types of services and have already proven popular (14.4 million portable satellite navigation systems sold in Western Europe in 2007 [144]). There is also an existing trend toward complementing satellite navigation related technology with local area wireless communications (by 2013 34% of all portable navigation devices will feature wireless cards [3]). An infrastructure based vehicular communications approach could allow early adaptors of wireless enabled satellite navigation devices to receive useful services from day one, regardless of the penetration level of the technology.

Value added services are more easily deployed than safety applications, but raise some misgivings of their own. The provision of services such as web browsing, video streaming, and Voice-over IP (VoIP) using vehicular communications sounds like an attractive proposition. However some of these services e.g. web browsing should not be used while driving (naturally any passengers in the vehicle can safely enjoy them). Also they do not harness the uniqueness of the vehicular space they are merely ports of existing services to the vehicular context. As such many potential value added services compete with services already available to a mobile device owner. There are value added services such as, location based

[1] Ranchi Traffic Jam, 2008, Mahadeo Sen, India, viewed 28 December 2009,*http://newswing.com/?p=3418*

services, which can take advantage of the vehicular setting to provide fresh applications for everyday use. Example services include - information provision (e.g. information on prices at nearby fuel stations, parking space availability) and e-payment (e.g. at fuel stations, carparks and toll stations). These types of services can be seen as being complimentary to traffic efficiency services.

This thesis focuses on the *provision of traffic efficiency services using an infrastructure based approach*. It describes *Smart City, a novel framework*, which purposes the use of wireless communication to make city life greener and more efficient. A major contribution to this framework is the *proposed intelligent traffic management module*. A route management service, which is powered by *a best route selection algorithm*, is put forward as a prototypical traffic efficiency service for this module. The novel aspect is that the algorithm minimises journey times and traffic congestion as well as fuel consumption and emissions. Testing has shown how the algorithm provides - shorter journey times, a reduction in fuel consumption and harmful emissions and also results in financial savings. We have proposed and implemented *an infrastructure-based communication scheme that enables prioritisation of services provided to vehicles*.

1.3 Problem & Solution

1.3.1 Problem Statement I

Problem There is insufficient road capacity and limited space in urban areas resulting in major issues with traffic congestion. Existing solutions are insufficient to solve the problem and congested roads are part of daily life in major cities. Additionally there is a need to reduce gas emissions and consequently fuel consumption due to both economic and environmental factors. In this thesis it is assumed that in order to solve the congestion problem traffic must be managed in a way which reduces journey time, fuel consumption and gas emissions. The thesis is concerned with traffic management solutions utilising vehicular communications which can solve the problem.

Potential Solutions Potential traffic management solutions based on vehicular communications can be categorised according to two main approaches centralised and distributed solutions. In a centralised solution roadside infrastructure is used and data is pooled and validated at a central location. These solutions may have issues with cost and scalability as considerable infrastructure must be installed and maintained. In a distributed solution vehicles exchange information amongst themselves for example a passing vehicle can give information on where it has been and information from other vehicles it has passed. While these solutions have a significantly lower cost and should scale well there may be issues with the accuracy of information. Distributed solutions can build up information about the "local" area but the centralised approach should be able to provide a more "global" outlook if necessary.

There are good real world examples of the successful application of both centralised and distributed approaches to solve traffic management problems. Air Traffic Control is a centralised system used in airports worldwide to separate planes in the controlled airspace, prevent collisions and generally manage and hasten the flow of traffic. The system tracks plane locations and provides information, instructions and support to pilots. The air traffic control system is a complex system which deals with large amounts of data - at the worlds busiest airports a plane movement (take off/landing) occurs at the approximate rate of one every 30 seconds and worldwide airport passenger numbers in 2009 were over 4.796 billion (Statistics: Airport Council International[1]). This safety critical system works extremely well with air travel being statistically one of the safest forms of travel for example in 2007, 44 people died in air crashes in the United States, while auto crashes killed 44,000, according to the National Transportation Safety Board (NTSB[2]). Packet routing in the Internet is an example of a distributed solution for traffic management. The Internet is a vast distributed network made up of voluntarily interconnected independent networks. There is no central governing body but the Internet is a global phenomenon with approximately 1.84 billion users as of 2009, according to the International Telecommunication Union[3] and is considered to be one of the greatest advancements of the 20th century.

Solution The proposed solution is a vehicular communications based traffic management system. The system optimizes the existing infrastructure utilisation and minimises time spent in traffic and fuel consumed by moving vehicles. This thesis describes the Best Route Selection Algorithm which achieves these aims. A centralised approach is taken. The primary reason for this was a desire to not only reduce individual journey times, fuel consumption and emissions but also to combat the congestion problem globally. In order to reduce congestion across the whole traffic network it was felt the global view of the network provided by a central system provided the best chance of success.

There are enormous challenges in developing a large scale centralised traffic management system based on vehicular communications for a large urban area. Even for a modest sized urban area such as the town of Cambridge (UK) population 100,000, there are 183,850 vehicles passing through it in the 12 hours from 07:00 to 19:00 on a typical day [27]. When a large metropolitan area is considered it is clear that harvesting traffic data from vehicles will yield vast volumes of data. Scaling the system to such a level would be a demanding undertaking and processing the volume of data and disseminating control messages in real-time is a mammoth task. Given the amount of detailed information on the location of vehicle owners gathered by such a system, the potential for abuse is worrying for example unwarranted surveillance. Therefore maintaining the privacy of users and consequently the security of the system is also a serious concern.

1.3.2 Problem Statement II

Problem The number of cars in a busy traffic network is very high while wireless communications network infrastructure have limited capacity and coverage. In the vehicular

[1] Airport Council International http://www.aci.aero/
[2] National Transportation Safety Board http://www.ntsb.gov/
[3] International Telecommunication Union http://www.itu.int/

context services which are desirable such as traffic safety, traffic efficiency and value-added services are of varied levels of importance to users.

Potential Solutions Given that the communications solution must support a centralised traffic management system it can be assumed that some form of infrastructure based solution is required. Potential wireless communications infrastructural solutions fall into a number of categories; ubiquitous roadside-vehicle communications (URVC), sparse roadside-vehicle communications (SRVC) and hybrid-vehicle communication systems (HVC). With URVC and SRVC systems communication occurs only between vehicles, with URVC providing full coverage and the latter providing coverage intermittently at communications hotspots. With these two approaches there is a trade off between coverage and cost. In the case of HRVC systems the range of an SRVC system is extended by the addition of ad-hoc communication between vehicles. This may provide a compromise between cost and coverage but increases the complexity of the system.

The infrastructure-based communications approach can be subdivided into two types; ubiquitous roadside-vehicle communications (URVC) and sparse roadside-vehicle communications (SRVC) discussed hereafter.

Solution The proposed solution enables the communication for very high numbers of vehicles via a sparse wireless network infrastructure and supports prioritization of data associated with different service types. This thesis presents the Prioritized Data Exchange Algorithm which supports differentiated communication between vehicles and roadside infrastructure points.

1.4 Contributions

This thesis contributes to the state of the art in the area of wireless vehicular networks by introducing three novel aspects.

- The Smart City Service Oriented Architecture is a framework which enables the provision of a wide set of services to suitable mobile devices, supported by wireless communications. The basic functionality includes data harvesting, and data processing to gather information to a geospatial database. This information can be accessed by a wide range of services which can be added in a modular fashion. Traffic management services are an example of typical services supported by this framework.

- The Best Route Selection Algorithm powers a route management service for vehicles, which fits into the Smart City framework. It deploys in a sparse roadside infrastructure-based vehicle communications network. The novel aspect is that it minimises journey time, traffic congestion as well as fuel consumption and emissions.

- The Prioritized Data Exchange Algorithm supports the route management service and other intelligent transportation services which rely on wireless vehicular communication. It applies to sparse road sparse roadside infrastructure-based vehicle communications network. The novel aspect is that it enables increasing number of vehicles and growing amounts of data to be exchanged while prioritising data delivery based on service importance.

1.5 Publications

The Publications arising from this thesis are:

- Kevin Collins and Gabriel Miro Muntean. TraffCon: An Intelligent Traffic Control System for Wireless Vehicular Networks, *In Proc. of the IET China-Ireland International Conference on Information and Communication Technologies (CIICT)*, Dublin, Ireland, August 2007.

  Summary: This paper introduces the concept of the centralised traffic management system and presents the system architecture.

- Kevin Collins and Gabriel Miro Muntean. A Vehicle Route Management Solution for Wireless Vehicular Networks, *In Proceedings of 27th IEEE Conference on Com-*

puter Communications (IEEE INFOCOM 2008): Mobile Networking for Vehicular Environments (MOVE 2008), Phoenix, AZ, USA, April 2008.

Summary: In this paper a route management solution for the traffic management system is presented. The solution combats the traffic congestion problem by seeking to optimize the usage of existing road capacity. It demonstrates that the solution increases usable road capacity and reduces traffic congestion. Journey times and fuel consumption are both shown to decrease for vehicles travelling on the road network.

- Kevin Collins and Gabriel Miro Muntean. Route Based Vehicular Traffic Management Solution for Wireless Vehicular Networks, *In Proceedings of 68th IEEE Vehicular Technology Conference (IEEE VTC 2008-Fall)*, Calgary, Canada, September 2008.

  Summary: The route management solution is further evaluated in this paper. A larger road network is used and how the solution achieves its improvements is examined. It is shown that journey time and fuel consumption are decreased because re-routing vehicles according to the solution increases their average speed while either reducing journey distance or not increasing it enough to have a negative impact. A reduction in braking or stop/start behaviour also adds to the increase in average speed and helps to reduce fuel consumption.

- Kevin Collins and Gabriel Miro Muntean. An Adaptive Vehicle Route Management Solution for Wireless Vehicular Networks, *In Proceedings of 68th IEEE Vehicular Technology Conference (IEEE VTC 2008-Fall)*, Calgary, Canada, September 2008.

  Summary: An enhancement of the best route selection algorithm which underpins the route management solution is detailed in this paper. Adaptivity is added to the solution allowing for further, the adaptive solution is designed to avoid the production of undesirable flash crowd effects. Further reductions in journey times and fuel consumption are shown for vehicles travelling on the road network.

1.6 Thesis Structure

The remainder of this thesis is structured as follows. Chapter 2 chapter provides background information on both traffic efficiency and vehicular communications. Chapter 3 provides an overview of important research works related to this thesis.

In chapter 4 the Smart City service oriented architecture is depicted and a route management service which fits within the framework of the Smart City architecture is presented. The best route selection algorithm is introduced for the first time as the driving mechanism behind the route management service.

In chapter 5 a sparse roadside infrastructure based vehicle communication architecture is described. The principle behind the communication is discussed as well as the need for prioritisation. The prioritised data exchange algorithm is introduced here for the first time as a mechanism for initiating and managing communication between vehicles and roadside infrastructure.

In chapter 6 the algorithms related to this thesis are outlined namely, the Best Route Selection Algorithm and the Prioritised Data Exchange Algorithm. In chapter 7 the results of experimental tests measuring the performance of these algorithms are presented and analysed. Finally in chapter 8 conclusions are drawn and potential future works discussed.

Chapter 2

Background

This chapter provides some historical background information on both traffic efficiency and vehicular communications. Traffic efficiency approaches proposed before the arrival of vehicular communications are examined. A number of interesting solutions, which do not utilise vehicular communications, are presented. Vehicular communications will play a central role in traffic efficiency going forward; they will provide solutions that assist, augment or supersede traditional approaches. In order to give some understanding on the background of vehicular communications we present some of the earliest works and standards currently under development in the area are discussed.

2.1 Traditional Traffic Efficiency Approaches

Even before the advent of the auto-mobile, traffic congestion had become a major problem in the largest urban areas. As congestion and traffic discord became a major social issue the authorities recognised that it was necessary to exert more control over the highways and byways. Governments in conjunction with urban planners and policy makers have attempted to manage traffic and reduce congestion with ambitious infrastructural developments and innovative policies for over a century and they continue to do so.

On 10 December 1868, the first traffic lights were installed outside the British Houses of Parliament in London, by the railway engineer J.P. Knight [112]. They resembled railway signals of the time, with semaphore arms and red and green gas lamps for night use. The gas lantern was turned with a lever at its base so that the appropriate light faced traffic. Unfortunately, it exploded on 2 January 1869, injuring the policeman who was operating it.

In the intervening years there has been a continued push for infrastructure and policy advancements that overcome traffic problems. Solutions for positively affecting the traffic system can be broadly categorised into three approaches - *Increasing Capacity, Improving Efficiency* and *Controlling the Demand* discussed forthwith.

2.1.1 Increasing Capacity

The concept of increasing capacity is as simple as it sounds and for the most part revolves around the construction of new roads and the expansion of existing road systems.

However a number of novel approaches have been taken in this area, including making more efficient use of ground space with dynamic lane allocation and better use of air space in urban areas via multi-level roadways.

Dynamic lane allocation has been implemented in many cities worldwide. The solution is based on a simple principle, i.e. rush hour is a predominantly one-way phenomenon. In the morning, inbound lanes are crowded and outbound lanes have only a small number of cars passing through them; in the evening the inverse applies. However, the number of lanes are allocated equally at all times, e.g. in an eight lane system there are four lanes each way. If the lanes could be dynamically allocated to match the traffic conditions then the effective capacity could be increased e.g. in an eight lane system in the morning six lanes inbound are provided and two outbound.

An existing example of how airspace can be used to provide additional road space by building multi-level roads can be seen in figure 2.1. However such systems have high construction costs and the on and off ramps can potentially be traffic bottlenecks if not managed correctly.

Figure 2.1: All three levels of Wacker Drive in Chicago[1]

2.1.2 Improving Efficiency

There are a great many areas where improved efficiency in the operation of traffic network can be sought e.g. access management, traffic signal timing and coordination, accident prevention, transit routing and scheduling, etc.

One active research area in access management is *ramp metering*. Ramp metering is the use of traffic signals at motorway on-ramps to control the rate of vehicles entering the motorway. The signals can be set for different metering rates to optimise motorway flow and minimise congestion. Work in this area includes [126, 121, 163].

Signal timing algorithms and the use of real-time data from mainline loop detectors for adaptive signalling has also been a popular research topic. Several adaptive traffic control systems have been implemented for intersections all over the world. Some of the most important ones include Split, Cycle and Offset Optimisation Technique (SCOOT) [20] and Sydney Coordinated Adaptive Traffic System (SCATS) [4].

2.1.3 Controlling Demand

Controlling demand approaches refer to, initiatives that reduce user demand for road travel or encourage users to satisfy their demand for road travel in a more efficient way e.g. mass

[1] Wacker Drive, 2005, wikipedia, US, viewed 28 December 2009, *http://en.wikipedia.org/wiki/File:Wacker_Drive,_3_levels.jpg*

transit and car pooling both reduce the number of vehicles on the road freeing up capacity.

Road pricing or congestion charging is one way drivers can be discouraged from using their vehicles unless absolutely necessary. By overcharging users of the road network in periods of peak demand traffic congestion is reduced. The technique is applied in a number of cities around the world including London, Stockholm, Singapore and Milan. A number of approaches can be taken when implementing congestion charging including; a cordon area around the city centre - there are charges for passing the cordon line, area wide congestion charging - there are charges for being inside an area and a city centre toll ring - toll collection surrounds the city centre.

The provision of free or subsidised public transport is another initiative that aims to persuade people to leave their cars at home. Such measures have been piloted with considerable success in various cities around the world, including Brussels [104, 38], Clemson, Lansing and Bloomington [114]. These initiatives assume the classic layout of urban areas with a central hub where the majority of shops and businesses can be found. The majority of people live outside of this area and commute inward. An alternative approach is to alter this dynamic, using new urban development patterns.

There has been a large amount of research done into how development patterns or land use affects the transport system, e.g. [96, 170, 7, 175, 149]. A variety of *urban planning policies* have been put forward to change the way in which development takes place. The resulting land use patterns aim to reduce the use of private vehicles and make public transit, walking and bicycles more attractive for some trips.

One proposed development pattern is "Smart Sprawl". Land uses, like large commercial centres, and transportation facilities like high capacity freeways, concentrate traffic. Low-density, dispersed land uses, on the other hand, spread traffic widely; they facilitate increased per capita vehicle use, but also decrease the overall density of vehicle travel and, hence, reduce congestion - this has been described as "Smart Sprawl" [154].

These traditional traffic efficiency techniques based on increasing capacity, improving efficiency and controlling demand are well established and do not use vehicular communications like the traffic efficiency solution proposed in this thesis. Many of these older solutions involve the construction of large scale civil engineering projects. In urban areas the available physical space places an upper limit on solutions of this kind for example there is a limit to how many additional lanes you can add to roadways in order to create additional capacity. Vehicular communication based Traffic Management applications can be a complementary solution to traditional traffic efficiency approaches. For example they can be used to; co-ordinate the best use of the existing infrastructure by vehicles, gather additional information on road to improve the efficiency of existing signalling infrastructure and gather data that can be used to make better informed decisions in the urban planning process. In the next section some of the early work, which opened up the area of research, now referred to as vehicular communications is discussed. The emerging standards in the field of vehicular communications are also presented, if vehicular communication based solutions to traffic efficiency are to become as widely deployed as existing solutions such as traffic lights are today then standards are necessary.

2.2 Vehicular Communications

Research in the area of vehicular communications (VC) has been active for over 20 years. This section provides a brief overview of some of the pioneering work in this area and the standards which have emerged for vehicular communications.

2.2.1 Pioneering Research

Much of the initial research into vehicular communications was focused on achieving short-range communication between two moving vehicles. This single-hop data transfer scenario posed enough of a challenge that multi-hop systems were not yet considered. There were only a small number of projects honing in on specific applications for vehicular communications. One such project was the Program on Advanced Technology for the Highway (PATH) [142, 164]. PATH focused on automatic vehicle control tackling issues such as cooperative driving and vehicle platooning and is still active today [141].

The applications being considered (e.g. cooperative driving) required exact knowledge of the location of neighbouring vehicles. However GPS was neither widely deployed or sufficiently precise yet. Consequently considerable effort was devoted to determining vehicle location and distances between vehicles as described by *Yashiro et al* in [181]. Wireless LAN had not been standardized at this point so various solutions were put forward at both the physical layer and the MAC layer.

At the physical layer a variety of solutions were put forward. A number of papers proposed optical solutions such as in a paper by *Fujii, Hayashi & Nagata* [60], where an infrared based communication system is presented. Other papers such as those *by Inoue & Nakagawa* [76] and *Mizui et al* [115] put forward solutions based on spread spectrum techniques.

A number of solutions were also proposed at the MAC layer. The Concurrent Slot Assignment Protocol (CSAP) was one such solution based on time division multiple access (TDMA) [107, 111, 110]. The Cooperative Optimized Channel Access for INter-vehicle communication (COCAIN) was another MAC protocol put forward by *Kaltwasser & Kassubek* [92].COCAIN uses carrier sensing approaches rather than slotting as found in CSAP. Different solutions based on the ALOHA protocol were also proposed [165].

In the intervening years a variety of communication technologies have been used to provide vehicular communications including; Optical (infrared) [99], Ultra-wideband (UWB) [50], Bluetooth [150] and Cellular [135]. However it appears that IEEE 802.11 (Wi-Fi) based solutions for vehicular communications are emerging as the most popular. That trend looks set to continue as the 802.11p [88] amendment to the standard designed to cater for VC, approaches publication. This and other standards will be discussed in the next section. Further background reading on vehicular communications can be found in recent survey papers by *Sichitiu and Kihl* [143] and by *Willke,Tientrakool & Maxemchuk* [177] and also a recent book on the topic by *Olariu and Weigle* [124].

2.2.2 Standards

The development of standards for vehicular communication was kick-started in 1999 by the allocation of 75MHz of Dedicated Short Range Communications (DSRC) spectrum at 5.9GHz by the U.S. Federal Communication Commission to be used exclusively for vehicular communications [55]. This DSRC spectrum ranges from 5.850 - 5.925 GHz and is divided into 7 10MHz channels. The EU has since followed this lead and allocated 30 MHz of spectrum in the 5.9GHz band [15].The term DSRC is used to refer to both the spectrum allocated for automotive use and the associated set of protocols and standards.

The FCC channel allocation for DSRC can be seen in figure 2.2; there is a control channel (178), two service channels (172 and 184) are reserved for safety critical applications and the remaining four channels are available for extended data transfer. Public safety applications and messages have priority in all channels [30]. DSRC is designed to support data rates of 6-27 Mbps and have a range of up to 1000 metres. Both vehicle to roadside and vehicle to vehicle communications are supported. The control channel is used to co-ordinate roadside to vehicle communications between roadside units and on-board units. The control process operates as follows[90]; roadside units announce 10 times per second over control channel 172 the applications they support, on which channels. Vehicle's on-board units; listen on control channel 172, authenticate RSU digital signatures, execute safety applications first, then switch channels and execute non-safety applications and finally return to channel 172 and begin listening again. The standardisation efforts in the area of DSRC are ongoing and involve standards organisations such as ASTM and IEEE.

The first efforts at standardising DSRC radio technology were begun by American Society for Testing and Materials (ASTM) resulting in the publication of a specifications document in 2003 [9]. In 2004 these efforts migrated to the IEEE 802.11 standard group resulting in the new amendment IEEE 802.11p Wireless Access in Vehicular Environments (WAVE) [73]. IEEE 802.11p describes how communications occur over each channel in the DSRC spectrum, but a complete communications system for WAVE needs to include support for multi-channel operations, security, and other upper layer operations [88]. These concerns are addressed by the IEEE 1609 family of standards for Wireless Access in Vehicular Environments [71, 72, 74, 70].

In Europe work is under way to standardise a communication architecture for intelligent transportation systems [97]. To achieve this goal a group of major European projects have

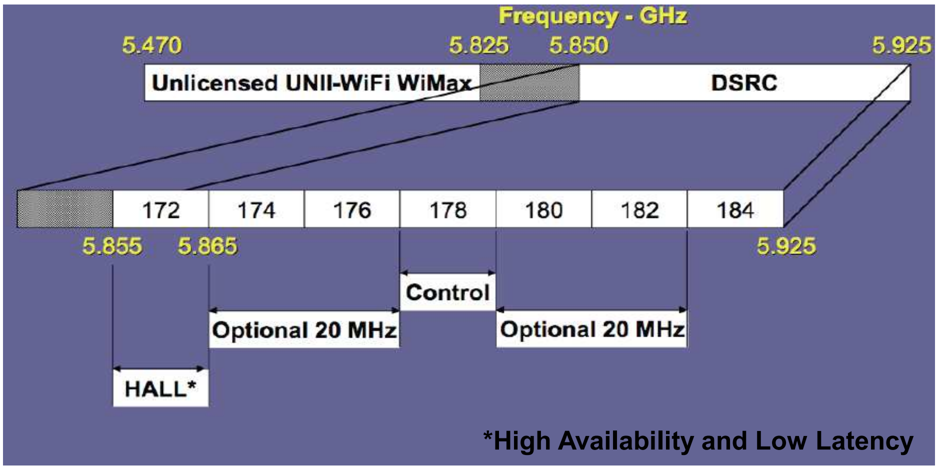


Figure 2.2: DSRC Channel Allocation [90]

been coordinated by the European Project COMeSafety [1]. Contributing projects include - Cooperative Vehicle-Infrastructure System (CVIS) [2], COOPERativ networks for intelligent road Safety (COOPERS) [3], Safespot [4] and Secure Vehicular Communication (SeVeCom) [5]. The standardisation efforts are summarised in [97] by *Kosch et al* and greater detail can be found in the public deliverables published online by the participating projects e.g. the European ITS Communication Architecture Overall Framework Proof of Concept Implementation published by COMeSafety [18].

The Transport Protocol Experts Group (TPEG) was founded in 1998 by the European Broadcasting Union to develop new protocols for Traffic and Travel Information, for use in the multimedia broadcasting environment. To date two standards associated with vehicular communications have been developed by TPEG. These are the TPEG binary data format, designed for transmission over DAB (Digital Audio Broadcasting) [129, 78, 79, 80, 81, 82, 77] and tpegML an XML implementation designed for use in editing systems and delivery

[1] COMeSafety: http://www.comesafety.org/
[2] Cooperative Vehicle-Infrastructure Systems: http://www.cvisproject.org
[3] COOPERS: http://www.coopers-ip.eu/
[4] Safespot: http://www.safespot-eu.org/
[5] SeVeCom: http://www.sevecom.org/

via the Internet and DVB (Digital Video Broadcasting) [130, 83, 84, 85, 86].

Further background reading on standardisation bodies and the technical committees, working groups etc. which may be relevant to standardising aspects, of vehicular communications (e.g. ISO Technical Committee 204 intelligent transport systems [2]) can be found in a recent standardisation overview published as part of the public deliverables of the COMeSafety project [139].

2.3 Chapter Summary

This chapter has examined traditional approaches to traffic efficiency i.e. current approaches that do not use vehicular communications. As vehicular communications will be an important technology for future traffic efficiency applications, some of the pioneering research was discussed and some standards currently under development in the area of vehicular communications were presented. This chapter also provides the historical context for the works more closely related to this thesis that will be discussed in the next chapter.

Chapter 3

Related Works

This chapter will present research work related to this thesis, which can be divided into a number of sections. First the solutions proposed in the general area of achieving traffic efficiency via vehicular communications are discussed. Then the proposed roadside-vehicle communication schemes capable of supporting traffic efficiency applications are described.

3.1 Traffic Efficiency

There are numerous solutions for improving traffic efficiency employing vehicular communications. These approaches to traffic efficiency can be allocated to three distinct categories, which are dealt with hereafter: *Autonomous Vehicle Systems, Traffic Information/Advisory Systems* and *Traffic Management Systems*.

3.1.1 Autonomous Vehicle Systems

Autonomous vehicle systems can provide traffic efficiency solutions by fully automating vehicles and thereby removing user responsibility for driving. There has been and continues to be a wealth of research in this area, some of the most important of which will be presented next.

The California Partners for Advanced Transit and Highways (PATH) program has been active in the area of intelligent transport systems (ITS) since 1986. The project has made a strong contribution in the area of autonomous vehicle systems with work on automated highway systems and advanced vehicle control and safety systems [141]. Grouping vehicles into platoons was seen as one way in which traffic efficiency could be improved. In 2003 PATH demonstrated three 18-wheel trucks and three buses operating in platoons and performing various automated maneuvers [51]. They have also worked on the automation of heavy duty highway snow-blowers [152]. The PATH programs contribution to the field of autonomous vehicle systems continues unabated and recent work includes field Demonstration and tests of automation of lane and docking maneuvers for a 60ft articulated bus [153]. This work on automation will continue with the development of a system that enables buses to negotiate docking stations, tolling lanes, and right-of-way lanes with precision that improves efficiency and quality in a number of transit applications [65].

Figure 3.1: Two unmanned vehicles participating in the DARPA urban challenge 2007[1]

Outside of the PATH programs efforts on autonomous vehicle systems some of the most celebrated work in the area has stemmed from the U.S. Defense Advanced Research Projects Agency (DARPA) urban challenge [140, 168]. The Urban Challenge features autonomous ground vehicles travelling through a mock city environment while performing all the complex maneuvers handled by humans in the real world e.g. merging into moving traffic, navigating traffic circles, negotiating busy intersections, and avoiding obstacles.

A large number of research teams have furthered autonomous vehicle systems through

[1] Stanford Racing and Victor Tango together at an intersection, 2007, DARPA, US, viewed 28 December 2009, *http://www.darpa.mil/grandchallenge/gallery.asp*

the DARPA initiative. Here some of the most successful will be discussed. Virginia Tech's victor tango research team developed an impressive autonomous vehicle called 'Odin', which utilizes sensor technologies such as computer vision, laser range finders, differential GPS and inertial measurement [11, 69]. A research team from Stanford University also developed a vehicle that could autonomously navigate an urban environment. The vehicle, named 'Junior', can chose its own routes, perceive and interact with other traffic, and carry out the actions necessary for urban driving e.g. lane changes, U-turns, parking, and merging into moving traffic [116]. The final competition stage of the DARPA Urban Challenge was won by a research team from Carnegie Mellon University whose autonomous vehicle, 'BOSS' proved most adept at navigating an urban environment [162]. Strong contributions in the area have also been made by research teams from Cornell University [113], the University of Pennsylvania [17] and MIT-Cambridge [101].

The field of autonomous vehicle systems could theoretically offer an alternative solution to the traffic efficiency solution proposed in this thesis. However, autonomous vehicle solutions are currently being employed in military scenarios where as in the context of the civilian traffic problem the use of autonomous vehicle solutions is very much future research. In the distant future autonomous vehicle systems may offer a solution to traffic problems but at present they are a number of significant problems associated with the technology. Firstly such solutions are prohibitively expensive for large scale deployment, even if costs can be reduced they must then overcome the challenge of user resistance to automation and meet the stringent reliability criteria expected of safety critical systems. Until these barriers have been removed they will remain restricted to the unmanned military type applications currently envisaged in projects such as the DARPA Urban Challenge [167].

3.1.2 Traffic Advisory/Information Systems

Traffic advisory/information systems are solutions in which traffic data is gathered and traffic information is disseminated to users, so they can make better informed in-vehicle decisions. The hope is that individual users can achieve transit efficiencies in terms of various vehicular or user related metrics such as road usage, fuel consumption etc. There are

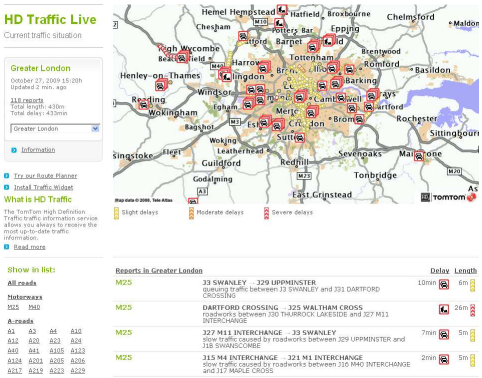


Figure 3.2: TomTom HD Traffic Information for London as seen online[1]

Figure 3.3: Vexia EcoNav's gear and accelerator pedal indicators[2]

both commercial solutions and research projects in this space, both of which are examined hereafter.

Since the launch of the 24th Navstar satellite into orbit in 1993, completing a network of 24 satellites known as the Global Positioning System (GPS) and the subsequent decision by the US Government to make the highest level of GPS accuracy available for civilian use [122] - in car satellite navigation systems have become widely deployed. In recent years manufacturers of satellite navigation (or sat-nav) systems have begun to augment their devices by allying the existing technology with wireless connectivity.

In 2009 the popular sat-nav provider TomTom [3] launched the TomTom Live connected service bundle. In the same year fellow market leader Garmin [4] launched a similar connected service named Garmin NuLink. In both cases connectivity is provided through partnership with mobile operators e.g. with AT & T in the U.S [156, 155]. The connectivity is used to provide traffic efficiency services for example TomTom Live features a service called HD Traffic [117] which allows fastest route selection using traffic information made available by tracking mobile phones and compatible TomTom devices to generate average speeds for sections of road. The information available can be seen on the TomTom website as shown in figure 3.2.

[1]www.tomtom.com/hdtraffic/

[2]Vexia EcoNav, 2008, Vexia, Spain, viewed 28 December 2009, *http://www.vexia.co.uk/images/productos/serie80/4/high/vexia_econav_view.jpg*

[3]TomTom: http://www.tomtom.com/

[4]Garmin: http://www.garmin.com/

[5]Vexia econav: http://www.vexia.co.uk/.

Figure 3.4: Google Maps Navigation - live traffic info [1]

Figure 3.5: Google Maps Navigation for Android - street view[2]

The traffic efficiency services found on today's commercial sat-nav devices are not solely aimed at reducing journey time. There are also devices that focus on reducing fuel consumption. One such device is the Vexia Econav [5], which aims to help users drive their vehicle in the most economical and ecological fashion. The user enters the make and model of their car and the unit tracks your speed acceleration and deceleration. It then recommends the best gear and engine revs at any given time using a gear and accelerator pedal indicator [145] as seen in figure 3.3. Dedicated sat-nav devices are no longer the only devices capable of providing this type turn-by-turn navigation. Take for example the smart-phone; the 3G communication technology comes as standard on Smart Phones, GPS has become a de facto standard feature and improved processing power and memory means navigation applications are commonplace on these devices. One such application Google Maps Navigation, a turn-by-turn navigation application for use while driving (similar to Garmin/TomTom type navigation), is included out of the box on smart-phones that run the Android OS 2.0 or higher. Using 3G communications live traffic information can be added to these application one example of this is the live traffic information for Google maps navigation as seen in figure 3.4.

[1]Google Maps Navigation, 2009, Google, USA,viewed 28 December 2009, *http://www.google.com/mobile/navigation/gallery/full/popular-search-layers.jpg*

[2]Google Maps Navigation, 2009,Google, USA, viewed 28 December 2009, *http://www.google.com/mobile/navigation/gallery/full/street-view3.jpg*

On top of these new innovations in commercially available systems there is also considerable research activity in the traffic information systems space. The C3 project at the AquaLab - Northwestern University works on a real-time information system that does not rely on any roadside infrastructure, but adopts a cooperative model cars collect information with their own sensors and via exchange with other vehicles they come across. Researcher involved in C3 have compared this approach with a centralised infrastructure based approach [125].

The German project NoW - Network on Wheels (whose partners included Daimler, BMW and Volkswagen) takes a hybrid approach to information dissemination in vehicular networks. The NoW system uses direct Wi-Fi communication among cars as well as between cars and road side communication equipment without the need for a coordinating infrastructure [56, 158].

At Rutgers University Disco Lab researchers are working on traffic information systems that use Wi-Fi based Vehicular Ad-hoc Networks. This work includes TrafficView, an application which both gathers and disseminates information about on the road vehicles, providing drivers with an extended horizon, i.e. a real time view of road traffic far beyond what they can actually see [37, 119]. Rutgers researchers have also developed the Vehicular Information Transfer Protocol (VITP) a location-aware, application-layer, communication protocol designed to support a distributed service infrastructure over VANETS [43, 58].

The CarTel project at the Massachusetts Institute of Technology (MIT) is another example of vehicles being used to both gather information and then deliver it via Wi-Fi. Work in this project includes Cabernet, a system for delivering data to and from moving vehicles using open IEEE 802.11 (Wi-Fi) access points encountered opportunistically during travel [52]. Researchers at CarTel have also proposed a mobile phone based traffic information system called iCartel[1]. iCartel is an iPhone app, which aims to reduce the time users spend stuck in road traffic by delivering traffic reports and calculating routes that experience the lowest congestion and delay. The application predicts traffic congestion by collecting data from mobile phones and the CarTel project has presented work specific to this area i.e.

[1] http://icartel.net/icartel-docs/

the VTrack system. The VTrack system estimates travel times using mobile phones and addresses issues related to energy consumption and sensor unreliability [157].

Rybicki et al also leverage mobile phones to create a traffic information system, they have proposed a peer to peer system, which uses cellular internet access [133, 134, 62]. Other research worth noting includes StreetSmart, a system that identifies and disseminates traffic patterns to users [44] and SOTIS, a system that distributes up-to-date travel and traffic information pertinent to a vehicle's locale [178].

Traffic advisory/information systems have some similarities to the traffic efficiency solution presented in this thesis. For example the instructions given by a traffic management system could be delivered in a similar fashion to the traffic advice given in a traffic advisory system such as that seen in figure 3.5. However Traffic advisory/information systems tackle the traffic efficiency problem in a different way to the traffic efficiency solution presented in this thesis. They are concerned with giving individual drivers information that helps them make their journey more efficient not optimising the whole traffic network. These systems keep drivers better informed about traffic conditions but there is no telling how the driver will interpret the information given. Consequently there is no guarantee such systems lead to more beneficial or optimal driving decisions.

Traffic advisory/information systems are concerned with getting the best result for an individual with little or no concern for the global ramifications whereas traffic management systems look to optimize the traffic system as a whole. One advantage of this approach is that either a distributed or centralised approach may be taken to data dissemination because there is no need co-ordinate the advice given at a global level. Provided enough information can be collected by users, on board devices using a distributed approach (e.g. a smart phone running a traffic information application) should be able to give traffic advice as well as those underpinned by a centralised system. The option to use a distributed approach can greatly reduce costs associated with infrastructure deployment and maintenance. Another disadvantage of this approach is that having a large number of users taking individually optimised, selfish traffic advice with no concern for the global outlook may have unpredictable

Figure 3.6: Parking is one area where researchers are looking for efficiencies[1] [2]

and potentially negative results for the overall traffic system.

3.1.3 Traffic Management Systems

Systems that actively control aspects of the traffic network in order to force member nodes into a behaviour that has some benefit to the system as a whole can be classified as traffic management systems. Elements of the traffic system that researchers have looked at managing using vehicular communications include traffic lights, speed limits, highway lane entry, parking spaces and road surface quality.

The Disco Lab at Rutgers University has proposed a number of traffic management applications. Rutgers researchers have developed an adaptive traffic light system based on Wi-Fi communication between vehicles and traffic lights, which improves on existing camera or sensor based approaches such as SCATS discussed in chapter 2 section 2.1.2 [64]. In addition to that work they have proposed an intelligent lane reservation system that would allow users to reserve entrance to a high-priority (fast) lane by paying a premium [75].

Other researchers have concentrated on parking space management. They aim to reduce the volume of traffic on the road by minimizing the time vehicles spend looking for spaces. *Caliskan et al* have developed a VANET approach to parking space discovery, which can be

[1]New York Sign, 2008, The Cartorialist, US, viewed 28 December 2009, http://www.thecartorialist.com/wp-content/uploads/2008/03/no_parking_sign.jpg

[2]Outdoor Car Park, 2008, Sinead Curran, Ireland, viewed 28 December 2009, http://sineadcurran.com/gallery/Outdoor Car Park_mid.jpg

used to direct users to a free parking space close to their destination [26, 25]. Researchers at Old Dominion University have proposed a parking management system that uses wireless network and sensor technologies. Users can reserve parking spaces before arrival and the problem of drivers parking in an improper fashion is eliminated [179].

The VGrid project at the University of California at Davis combines VANETS and grid computing to enable intelligent transport applications. The work of these researchers aims to improve traffic flow and to examine the potential of traffic alerts and dynamically varying speed limits based on the local density of traffic [34, 31, 100]. Poor road surface conditions (e.g. potholes) is another issue that affects traffic efficiency; *Eriksson et al* from the CarTel project at MIT have developed a vehicular communications solution to detect and report deteriorations in the road surface [53]. Such a system would allow civil authorities to better manage their road maintenance activities. Rail intersections have been identified as a traffic system element requiring improved controls by *Hartong et al*. They have proposed a system of train - vehicle communications to manage their interactions at road and rail intersections [67].

Cooperative Vehicle-Infrastructure Systems (CVIS) [1] is a European research project, aiming to design, develop and test the technologies needed to allow cars to communicate with each other and with the nearby roadside infrastructure. CVIS researchers have published as part of their public deliverables use cases which can take advantage of this work including a traffic management application called CURB [35]. Another European research initiative - Aktiv[1] which stands for ”Adaptive and Cooperative Technologies for the Intelligent Traffic” has similarly identified and outlined a traffic management use case for vehicular communication. Aspects of this proposal include cooperative traffic signalling and anticipatory and cooperative driving.

Individually the traffic management systems described above provide a benefit in some area of traffic management e.g. traffic light optimisation or parking space management. Unfortunately it is difficult to justify the cost of deploying such complex systems for a single use. However they share some basic principles i.e. the use of vehicle communications to

[1] Adaptive and Cooperative Technologies for Intelligent Traffic,
http://www.aktiv-online.org/englisch/projects.html.

gather data from vehicles and to then provide a traffic management application by delivering instructions to the vehicle or with infrastructure e.g. traffic lights. This means there is a need for a framework like the one described in this thesis, which allows these and other applications to be combined to form a more comprehensive traffic management system. The solutions could be combined with and complement the traffic management service described in this thesis in the Smart City architecture proposed.

Road pricing or congestion charging is one way drivers can be discouraged from using their vehicles unless absolutely necessary. A lot of work has been done in this area and research is ongoing, e.g. [16, 169, 24, 109]. London has the largest and most high profile implementation of congestion charging and the initiative has been broadly considered a success [138]. Congestion charging was introduced to London in 2003 and there have been annual reports on the impact since its introduction. These have shown that in the first 3 years the scheme yielded reductions in congestion of between 20-30%, as predicted. However in 2006 this dropped to 8% and still further to 0% in 2007 and the first quarter of 2008. However, the drop off in performance has been attributed to a range of interventions and incidents that have removed effective capacity from the central London road network [59]. It has also been shown that the scheme has had no negative impact on the local economy.

3.2 Roadside-Vehicle Communications

A comprehensive taxonomy of vehicular communications is provided by *Sichitiu and Kihl* [143] as seen in figure 3.7. There are three major types of systems: *Inter-vehicle communications systems (IVC)*, *roadside-vehicle communication (RVC) systems*, and *hybrid-vehicle communication systems (HVC)*. With IVC systems communication occurs only between vehicles, there is no wireless communications infrastructure needed. In the case of RVC systems all communication occurs between roadside infrastructure and vehicles. HVC systems combine vehicle-to-infrastructure communication and vehicle-to-vehicle communication. *This thesis focuses on roadside infrastructure-vehicle communications, only.*

The infrastructure-based communications approach can be subdivided into two types;

ubiquitous roadside-vehicle communications (URVC) and sparse roadside-vehicle communications (SRVC) discussed hereafter.

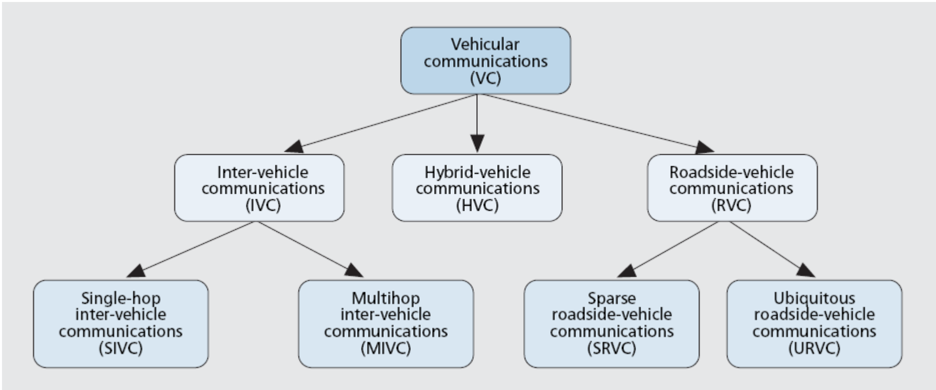


Figure 3.7: A taxonomy of vehicular communication systems [143]

3.2.1 Ubiquitous Roadside-Vehicle Communications

Ubiquitous roadside-vehicle communications assume that vehicles have always on connectivity anywhere on the road network. The large scale infrastructure necessary to provide an entire road network with high speed communication would require significant investment. However by having a full communications infrastructure, a large set of applications not enabled by other communication solutions are made possible.

One approach to ubiquitous roadside-vehicle communications is to use existing cellular mobile networks (e.g. GSM, GPRS and UMTS). *Santa et al* have proposed a cellular networks-based approach to vehicular communications [136, 135, 137]. They demonstrate a UMTS-based approach, which enables both vehicle-to-vehicle and vehicle-to-infrastructure communication. As part of the Network on Wheels project, *Wewetzer et al* have evaluated the relative strengths of UMTS-based solutions versus Wireless LAN for vehicular communications [176]. Both approaches are shown to have their merits; Wireless LAN is shown to be capable of supporting significantly higher data rates for short range single hop communication scenarios, whereas UMTS offers a reasonable data rate irrespective of distance. In [173] *Wang et al* demonstrate a method of increasing data rates when using hybrid cellular Wi-Fi networks for vehicular communications. The increases are achieved by trunking multiple cellular channels over an IEEE 802.11b vehicular ad-hoc network. Aside from limited data rates the main concern with approaches based solely on existing cellular networks is payment, as these use the licensed spectrum. The payment model that has been used by existing commercial services based on cellular networks such as TomTom Live and Garmin NuLink, is that of a subscription where the user pays for usage and data transmission.

There are full infrastructure solutions that reside in the unlicensed spectrum e.g. cellular Wi-Fi and Wi-Fi mesh networks. A number of municipalities have deployed large scale Wi-Fi networks for the provision of wireless services over the entire region including; Corpus Christi, Texas (population 300,000) and The City of Westminster, London (population 181,000) [61]. In Corpus Christi a citywide Wi-Fi mesh network was installed at a cost of $7 million. The wireless network is used to provide internet access to citizens as well as enabling new services such as automated meter reading (electricity) and mobile public safety data access for emergency service vehicles [10]. This survey paper [6] by *Akyildiz & Wang* provides an excellent introduction to wireless mesh networks. It has been proposed that mesh networks could provide the solution to the vehicle communications needed for intelligent transportation systems e.g. by *Bruno et al* part of their survey paper on mesh networks [21] and by *Huang et al* who propose a Wi-Fi mesh network-based solution for the implementation of intelligent transportation solutions.

A sparse roadside-vehicle communications approach is used in this thesis so these ubiquitous roadside-vehicle communication solutions represent competing approaches. Ubiquitous roadside-vehicle communications have some advantages for example the always on connectivity afforded by having a full communications infrastructure means it can support certain application that a sparse roadside-vehicle communications approach can not. One concern however with such solutions is that they require a massive investment in infrastructure before any benefit can be realised. For this reason they may be prohibitively expensive in many locations.

Figure 3.8: Roadside e-tolling gantry[1]

Figure 3.9: In car e-tolling RFID[2]

3.2.2 Sparse Roadside-Vehicle Communications

In the case of sparse roadside-vehicle communications, vehicles communicate data with intermittently available roadside infrastructure in order to provide services. These can be placed in two categories according to the communications involved -

- *Stand alone services* - communication occurs solely between the vehicle and a roadside unit, which often serves a single purpose e.g. a petrol station or a speed limit sign advertising its existence.
- *Integrated services* - The vehicle communicates with roadside units who share a common server via backhaul communications e.g. backhaul communications carry data from roadside units to a traffic control centre to enable the provision of traffic efficiency services across a city.

The deployment of stand alone services is relatively straight forward as it only involves one roadside unit operating in isolation. In fact e-tolling using in car RFID [132] tags is one example of this type of communication, which is widely deployed as seen in figures 3.8 and 3.9. However they are a variety of challenges that must be met when providing integrated services over a sparse roadside-vehicle network. We will discuss the research in this area ,which is commonly referred to as Vehicle-Infrastructure Integration (VII), hereafter.

In the US, the VII California project is one of the leading research efforts in the area of Vehicle-Infrastructure Integration. Researches working on the VII California project have installed a large scale test bed in Northern California using communications technology based on the WAVE trial standard [87]. The test bed has allowed field measurements of vehicle communications [41], the identification of issues associated with deployment [42] and the prototyping of applications e.g. collision avoidance at intersections [40]. The Japanese project SmartWay has demonstrated a number of vehicle-infrastructure integration applications including in car signpost information, travel times, merging assistance and parking lot payment [87]. The system utilises communication technology which is widely available due to two legacy systems which are deployed in Japan i.e. the Vehicle Information and Communication System (VICS) which uses 2.4GHz infrared communication and is owned by the Japan Road Traffic Information Centre, and the second system is a 5.8GHz DSRC-based Electronic Toll Collection (ETC) [123, 8, 91].

DOME the Diverse Outdoor Mobile Testbed project [148] at the University of Massachusetts, Amherst employs a vehicular network called DieselNet [23]. The test bed supports research ranging from infrastructure-based networking to sparse and dense ad-hoc networks. A diverse selection of communication technologies are supported including Wi-Fi, 3G, and GPRS. The vehicular network consists of 40 buses and the buses have both 3G and Wi-Fi capabilities. Passengers or other buses can establish Wi-Fi connections with a bus and access to the Internet is provided via 3G [147]. The test-bed is being used for a variety of research initiatives including; improving energy efficiency for battery operated nodes [13] and routing protocols [12].*Wang et al* have worked on a vehicle-infrastructure integration system based on IEEE 802.11b for the purposes of collision prevention [171]. More recently they have begun working with IEEE 802.11p and have proposed some enhancements to the standard [174].

[1]ERP Gantry, 2008, wikipedia, Singapore, viewed 28 December 2009, http://upload.wikimedia.org/wikipedia/commons/7/7a/ERPBugis.JPG

[2]E-ZPass RFID Tag, 2006, Computer Desktop Encyclopedia, US, viewed 28 December 2009, http://www.yourdictionary.com/computer/rfid-tag

3.3 Chapter Summary

This chapter examined the research related to the work presented in this thesis. First the general area of achieving traffic efficiency via vehicular communications was examined. Then roadside-vehicle communication schemes capable of supporting traffic efficiency applications were discussed.

Chapter 4

Traffic Efficiency System Architecture

In this Chapter a generic architecture for the provision of 'Smart City' services is introduced. The term 'Smart City' is used to describe an urban environment where a wide range of elements are enabled for communication with the aim of providing new services to suitable wireless devices. These services may be provided anywhere in the city, at home, in the high-street, in the car or on the bus. A specific use case for the generic architecture is described in the shape of a traffic management module that provides traffic efficiency services. A specific traffic efficiency service is proposed in the form of a novel route management service. The route management service minimizes vehicle journey times, fuel consumption and emissions while also minimizing the vehicle's effect on traffic congestion. The architecture of the route management service is presented and discussed in this chapter.

4.1 Smart City Service Oriented Architecture

In a smart city it is possible to create a host of new services by enabling previously passive elements of a city for communication. In figure 4.1 we can see examples of this for the home, on the high-street and for urban transport. If appliances that were previously non communicative are enabled then new services can be envisioned in the home for example

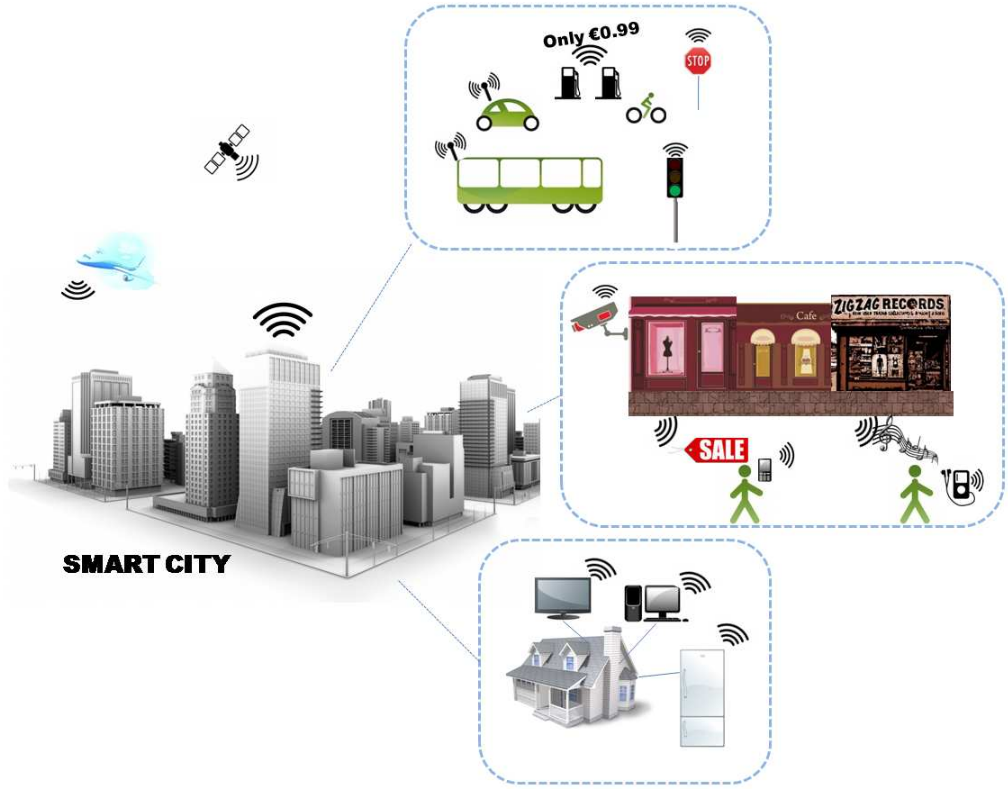


Figure 4.1: A range of elements communicating in a smart city

a fridge that can automatically order essential household items when stocks are low. When new elements of the high street are activated, services like automatic updates, rewards and incentives may be provided by your favourite retailers when you are nearby, for example receiving bonus music content from your favourite record store. By activating vehicles and other elements of the traffic network new services can be created for example information services such as a nearest/cheapest gas station recommender service or safety services such as intersection collision avoidance.

The Smart City system has a client-server architecture. City wide, elements of the urban landscape act as client nodes and communicate with the server in two asynchronous modes: information gathering and service provision as shown in figure 4.2. These modes will be outlined in greater detail in the succeeding sections.

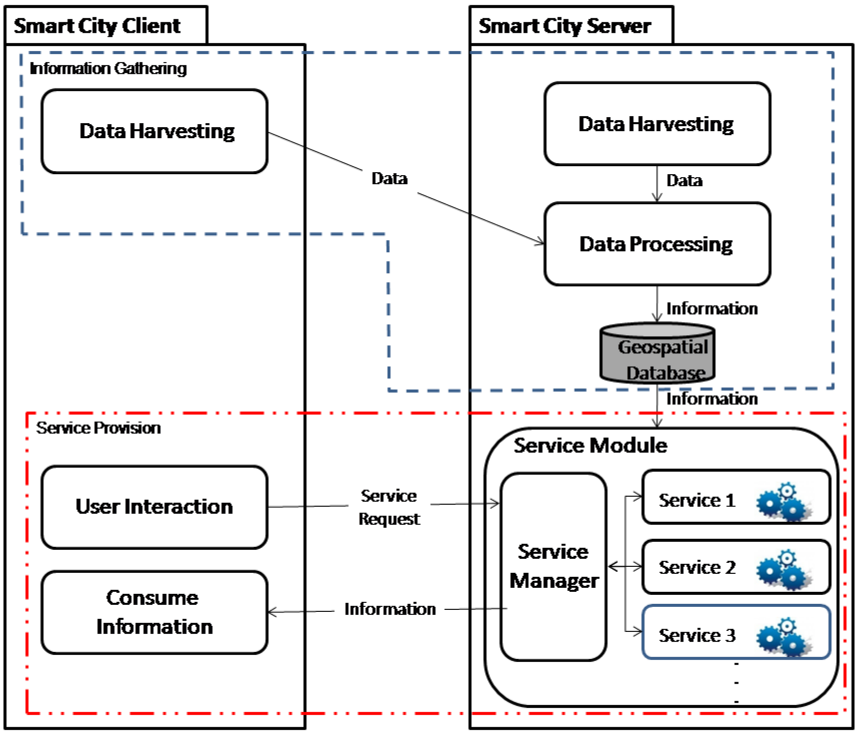


Figure 4.2: Smart city architecture

4.1.1 Information Gathering

During information gathering all nodes in the system collect useful temporal and spatially referenced data e.g. temperature, air quality, traffic conditions (i.e. Data Harvesting). This data is filtered, aggregated and refined to generate precise information regarding the state of the urban environment (i.e. Data Processing), which is stored in a suitable database. The communication for the information gathering process is not necessarily time critical, as depending on the information type it may not need to be up to the second to be valid. However, some threshold on the age of the information or separation of live information and historical information is required. These considerations also make up the data processing stage. In this phase inter-node communication may be used to employ techniques such as data aggregation in order to reduce the load on the network. It would be possible for simple client nodes to not avail of the service provision mode and only operate in information gathering mode. These can be seen as ”helper nodes” that provide useful data for the services more sophisticated nodes can avail of.

4.1.1.1 Data Harvesting

In order to harvest data relating to the state of the physical world networked sensors can be used. To build a picture of a large environment such as a city a large array of sensors must be used. Individual sensors may capture a single localized measurement such as temperature but a network of sensors may be combined to achieve a larger sensing task for example providing a weather report for the entire city. Mobile sensors are useful as they move about the environment and consequently can capture data about a wider geographical area. This is one of the reasons why vehicles make such a useful sensor platform. However this mobility also provides some challenges for networked sensors as wireless communications must be used to transmit data while on the move.

It is assumed that any networked sensors used are fit for purpose i.e. they take a set of measurements that provides the information needed to perform a prescribed task, this may be to detect a certain event or make a certain decision for example. That they are fit for purpose also means not only do the sensors take the appropriate measurements they do so with an acceptable level of accuracy i.e. they are correctly calibrated. If the information is to be geospatialy referenced (associated with a certain location at a certain time) then the sensor must associate a time and location with each measurement taken. A clock is required to time-stamp data. How location is ascertained depends on whether the sensor is fixed or mobile. A fixed sensor can be in a known location so if a unique identifier is supplied for the sensor then this may be mapped to a location. To provide a location solution for a group of fixed sensors a database table may be populated with the unique identifiers of the sensors mapped to their locations (this is a good example of the type of data harvesting that occurs on the server side). In the case of a mobile sensor location is variable so a suitable localisation solution must be employed to location-stamp data, GPS is one potential localisation solution.

Given that the networked sensors can capture useful data the final step of data harvesting in the outlined client-server architecture is to transmit the data back to the server. A sensor must take measurements at a suitable sampling rate in order to generate useful data. A networked sensor does not simply sample real world phenomena, it is a computer capable of filtering, combining and sharing its sensor readings with any internet equipped endpoint. These types of sensors can store and forward data when required, depending on the application requirements the communication policy of such sensors may vary. For example if the intention is to generate a monthly power consumption report for a building then sensors that measure the buildings power consumption need only forward data approximately once a month (given there is sufficient memory to store data) and a delay of a day or more may not be critical. When data is forwarded to a server with the intention of providing services in real time, then depending on the service there may be strict time constraints on forwarding data and delays of less than a minute may be critical. Sensors must have sufficient memory to store data until it is to be sent or else data will be lost.

The level of traffic generated will influence the choice of communications technology. To illustrate this take a networked sensor that samples at 1Hz generating 1kb of data for each sample. This sensor generates 1kb of data a second. Given that the application using the data requires that the data be at most approximately 5 minutes old, the networked sensor must send 300kb of data every 5 minutes. This is a relatively low data rate and if wireless communication is necessary there are a number of different technologies that can satisfy the communication needs depending on the deployment scenario. However sensors could potentially require very high data rates for example if the sensor was a high definition IP camera and the application was to provide a live video feed the data rates could be of the order of tens of megabits per second. Such a scenario would require a wireless communications technology with a suitably high data rate for example IEEE 802.11 a,g or n. A generic networked sensor for deployment on a mobile platform can be seen in figure 4.3 it contains all the attributes discussed sensor instrumentation, a clock, a localization solution, memory, a wireless communications solution and a processor to control their operation.

Any techniques that can reduce the traffic generated by Data Harvesting are valuable. This means a reduction in the amount of data communicated and less communication means less power consumption and potentially less network infrastructure, which can make for significant financial savings. One way of reducing traffic is to eliminate data redundancy. Data redundancy can be easily removed from fixed sensor deployments with sensible sensor

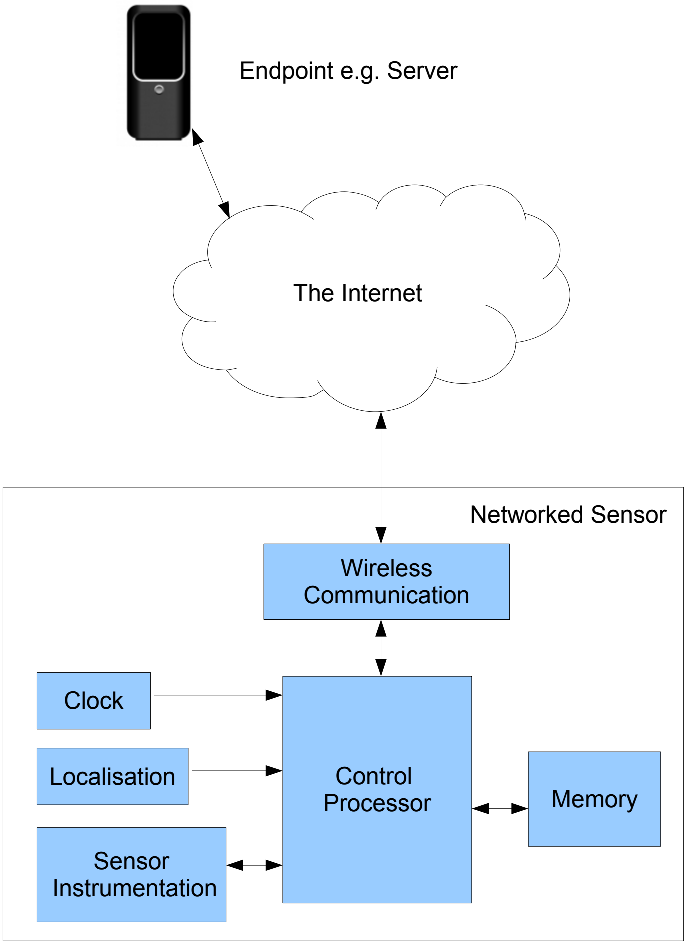


Figure 4.3: Generic networked sensor

placement. Fixed sensors should not necessarily have to take overlapping measurements for example if the purpose of a group of sensors is to measure room temperatures in a building then there should be only one sensor per room. Eliminating Data redundancy is a far more significant challenge in a mobile sensor scenario. Take a scenario where all vehicles in a city carry air pollution sensors, neighbouring cars stuck in traffic will take very similar measurements and data redundancy is likely. There are techniques that eliminate this type of redundancy such as data aggregation. In the case of mobile sensor nodes, neighbouring nodes could share there data by forming an ad-hoc network. One node can then be nominated to aggregate this data and forward the combined measurement to the server, reducing data redundancy.

In order to support a city scale deployment of networked sensors that report back to a central server, a suitable network topology must be employed. Given the amount of traffic generated by a city full of networked sensors there is a major challenge in designing a network topology that scales. For fixed sensors wired and wireless solutions or a combination of both may be equally valid depending on the scenario. One example of a wireless solution for deploying networked sensors in buildings is to use the ZigBee communications technology with a mesh network topology. Such systems have been used to automate light and heating controls in buildings and reduce power consumption [48]. The data from such a sensor array could be sent over the internet to a smart city server using a wired connection. For mobile sensors a suitable wireless infrastructure must be provided to allow communication back to the server. Different topologies may be used, provided the network has sufficient capacity to support the number of sensors deployed. Potential solutions include a cellular network topology using a technology such as 3G or a wireless mesh network topology using Wi-Fi technology. Once Data has been successfully communicated to the server then additional Data Processing may take place.

4.1.1.2 Data Processing

The processing techniques employed to generate useful information from harvested data are vitally important. Valuable data may be rendered useless if the analysis employed is not fit for purpose. The earliest attempts to measure the ozone layer using NASA satellites can be taken as a case in point. In 1985 British scientists reported a hole in the ozone layer of the earth's atmosphere over the South Pole. This news was disturbing, because ozone protects us from cancer-causing ultraviolet radiation. The report was at first discredited, as it was based on ground instruments and more comprehensive observations from NASA satellite instruments in place since 1979 had shown nothing unusual. The NASA satellite had for some reason not captured this plunge in ozone levels. Forced to revisit their data NASA discovered that they had did have data indicating ozone loss but had overlooked it. Where did they go wrong? Data processing - the satellite supplied scientists with information far faster than they could analyse it. To deal with the rapid influx of data, a data processing program had been set up to filter out all measurements below or above cut-off values that were considered to be impossible for actual ozone measurements. This program was based on the assumption that these ”impossible” measurements were due to instrument malfunctions [160]. The NASA scientists suppressed outliers as protection against errors in the data but in fact suppressed vital information which could have indicated the onset of a hole in the ozone the size of the United States. As this example of the hole in the ozone illustrates poor procedure in data processing such as suppressing an outlier without investigating it can conceal valuable information.

The first step in data processing is data validation - the process of ensuring data is clean, correct and useful. Achieving accurate validation is an important challenge, as data that is in error must be removed or it will lead to bad information. However as we have seen if the data validation process is flawed then it will suppress data in error and also lead to bad information. Depending on the type of network sensor providing data a set of validation rules will be used to test the correctness of data. A rule may be as simple as checking that a measurement falls within a certain range for example a length measurement should not be a negative value.

The next step is information generation. Information generation may be relatively simple such as the combination of a number of sensor measurements using a formula to calculate a new value or generate an average measurement or more complex such as event detection. Based on the arrival of certain data values an event in the real world may be detected, for example data arrives indicating a particular vehicle was travelling at a certain speed in a certain location. Given a database of speed limit information, if this speed exceeds a certain value, then it is possible to detect a vehicle breaking the speed limit. This information may be placed in a database table and used by an application that automatically administers speeding tickets. Any information generated must be stored so that it can be easily retrieved by services.

4.1.1.3 Geo-spatial Database

All the information generated must be stored in a secure and readily accessible fashion so that it can be retrieved quickly when needed by a service. In the Smart City Service Oriented Architecture information is shown being stored in a geo-spatial database. This is a spatial data infrastructure where spatio-temporal (space-time) location is the key index variable for all other information.This is the correct data infrastructure for a system that provides location based services. This type of database is used by other geographical information systems for applications such as remote sensing, urban planning and in-car navigation.

4.1.2 Service Provision

In order to facilitate the provision of new services, service modules attach to the geospatial database. A suite of services can be provided using services powered by algorithms that use information from the geospatial database as inputs. Client nodes that have a user interface that allows them to avail of services can make service requests to the service module, which are handled by a service manager. Once a client node is running a particular service the necessary information will be sent in a timely fashion to allow that service to operate. The time sensitivity of these communications would depend on the nature of the service.

The services that can be provided to a client depends on the data being collected around the city from other clients and by that client e.g. to provide a location based service where a client is told when it is within one hundred metres of another known client, both clients must be providing location information in the information gathering mode. It would also be possible to provide certain services to what can be described as ”selfish” clients who do not participate in the information gathering phase.

One desirable set of services, which could improve the standard of city life, are traffic efficiency services. In the next section an example of how a suite of such services can be provided using the smart city architecture is given. The traffic efficiency services are provided by deploying an intelligent traffic management module.

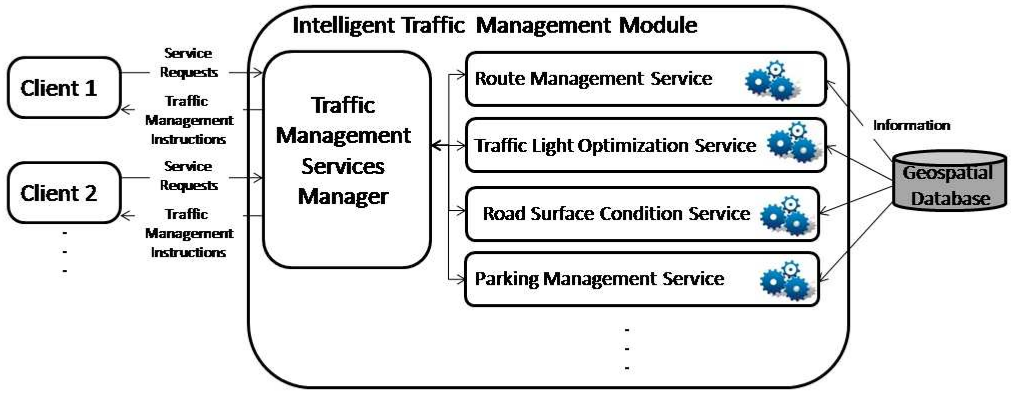


Figure 4.4: The Intelligent Traffic Management Module

4.2 The Intelligent Traffic Management Module

Traffic efficiency services can be provided in the smart city context with the deployment of an intelligent traffic management module, as shown in figure 4.4. Such a module would require that information on traffic conditions is obtained in the information gathering phase. Vehicles are one type of client that could contribute by acting as probes measuring e.g. the average speed on individual sections of road.

Given that the necessary information is available to the traffic management module a range of traffic efficiency services powered by some underlying algorithms may be provided. In figure 4.4 we include some example services based on algorithms that exist in research; traffic light optimization [64], parking space management [26, 25] and a road surface condition reporting service [53]. The final service shown is the route management service that is proposed in this thesis and discussed in greater detail in the next section. The target client may vary in each case e.g. vehicle drivers avail of the route and parking management services, traffic lights are the target client of the traffic light optimization and the public office responsible for road repair can avail of the road surface condition report service.

Finally the new services that can be provided in any module are limited only by the data being collected around the city. Take the example of the road surface condition reporting service found in [53]. In this case vehicles collect data from vibration and GPS sensors this data is processed to give information on the location of potholes and other road surface anomalies. All this means that a road surface condition reporting service can be provided, allowing the road infrastructure to be maintained in a more efficient manner. In order for other new services to be provided additional information must be acquired and this may mean the need for new sensor technologies, examples of these from research include lane detection [36, 95, 45] and pedestrian detection [118, 28]. In the context of the traffic management module if vehicles could identify their location at the lane level, then this allows traffic to be managed at the lane level or lane related safety applications, such as lane merging assistance, to be developed. With pedestrian detection capabilities, safety applications that reduce pedestrian-vehicle related injury can be introduced.

4.3 The Route Management Service

The operation of the route management service is expanded upon in figure 4.5. The service is powered by a best route selection algorithm, which is described in detail in section 6.1. The vehicle initiates the service by sending its original destination and desired destination in a service request. From there the route management service sends optimal route instructions based on information made available by other clients on current road traffic conditions e.g. other vehicles gather information on the state of the road traffic network such as the time taken to traverse road segments. The client vehicle provides updates on its location so instructions given can adapt to any deviations to the instructions given as well as changes in traffic conditions. The communications involved are time sensitive as instructions will only be valid within a certain time frame.

4.4 Chapter Summary

In this Chapter a generic architecture for the provision of 'Smart City' services was outlined. A specific use case where the aim is to provide traffic efficiency services is described.

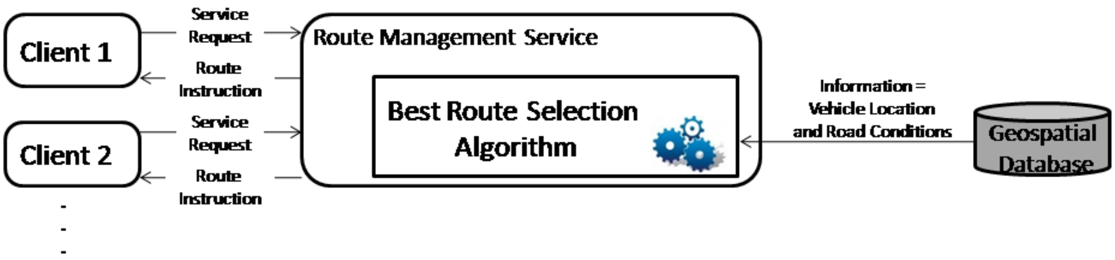


Figure 4.5: The Route Management Service

The traffic efficiency services are provided by a traffic management module for the smart city architecture. A specific traffic efficiency service is proposed in the form of a route management service. This service is powered by a best route selection algorithm, which is outlined in full in section 6.1 of chapter 6 Algorithms.

Chapter 5

Roadside-Vehicle Communications System Architecture

In order to provide traffic management services for vehicles as described in the previous chapter, communication between vehicles and a central system is required. In this chapter the basic communications approach for such a system is discussed and a specific architecture is presented. This architecture is for a sparse roadside-vehicle communications approach, which meets the communication needs of the system.

5.1 Basic Communications Approach

In order for the central system to provide services to vehicles they must be able to send requests to the server and the server must be able to respond as shown in figure 5.1. Vehicles can essentially be considered as mobile network nodes (vehicles may be parked but for the most part will be on the move when active) therefore, communications cannot be wired they must be wireless. Given a centralised system there must be some fixed network infrastructure to allow vehicles to communicate back to the server. If the system was a distributed one then a fully ad-hoc vehicular network would be a potential solution. The system is however centralised so vehicles communicate back to the server via some form of wireless infrastructure as shown in figure 5.2.

The wireless infrastructure which supports the necessary communications could take a variety of forms and use a mix of technologies. Solutions may be divided into three basic approaches Ubiquitous Roadside-Vehicle Communications, Sparse Roadside-Vehicle Communications and Hybrid Vehicle Communications. These approaches are explained in the following sections.

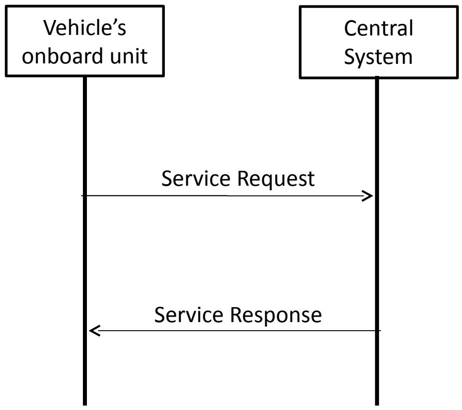


Figure 5.1: Vehicles request services from the central system

Figure 5.2: Vehicles and central system communicate over wireless infrastructure

5.1.1 Ubiquitous Roadside-Vehicle Communications

This approach uses a dense deployment of infrastructure in order to provide full coverage across the city. This blanket coverage makes wireless connectivity available to vehicles anywhere in the city. Vehicles can contact the central system over the Internet whenever or wherever it is necessary. The approach can be likened to the way in which mobile phone networks provide voice and data services across an entire country. There are a number of different ways this type of blanket coverage can be provided. For example a cellular network topology could be used with 3G or WiMax communication technologies, another alternative is a wireless mesh network topology with Wi-Fi as the underpinning communication technology. A cellular network approach can be seen in figure 5.3, a dense deployment of radio towers is used to cover the city in a similar fashion to a mobile phone network. This solution is expensive but enables the largest set of applications as a result of the always on connectivity.

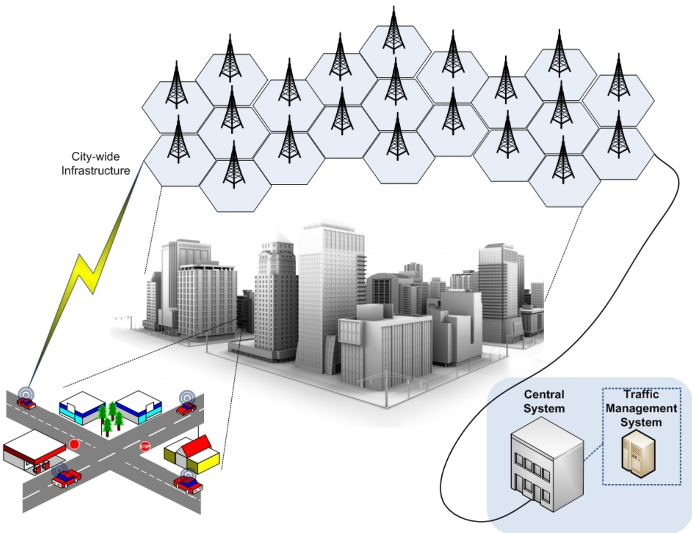


Figure 5.3: Citywide infrastructure lets vehicles connect to central system anywhere

5.1.2 Sparse Roadside-Vehicle Communications

In this approach communication hotspots are scattered across the city to give partial not full coverage with connectivity available intermittently when vehicles pass through a hotspot. Roadside units are deployed to create these hotspots that allow communication with the central system as seen in figure 5.4. This solution can be considerably less expensive than a full infrastructure approach but may not enable as many applications as connectivity is not always on. There are some additional challenges associated with this approach, most of which are related to the potentially short communication window vehicles may have with the roadside unit. Take for example a vehicle travelling at 100 kilometres per hour passing a roadside unit that has an effective range of 100m. This vehicle has a window of 3.6 seconds with which to talk to the central system via the roadside unit. In this time the vehicle must associate with the hotspot and send/receive a useful amount of data. Considerably lower speeds and stoppages due to traffic are likely in a city especially with heavy traffic but such a short connection window could be possible. As a result achieving fast association and high data rates are important challenges for this approach. A number of different communication technologies could be used for these hotspots including Wi-Fi, Bluetooth, Ultra-Wideband and ZigBee. The effective range, association time and throughput of the wireless communications technology are very important considerations as they affect how much data may be passed between vehicle and the central system during the connection window.

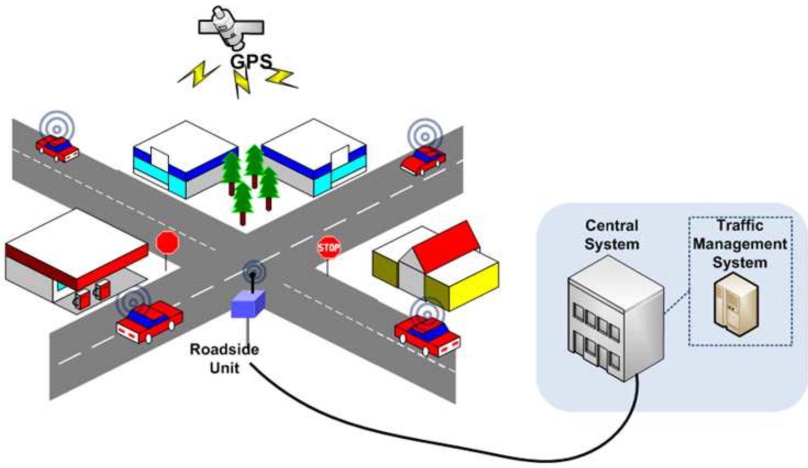


Figure 5.4: Communication between vehicles and a central system via a roadside unit

5.1.3 Hybrid Vehicle Communications

This approach is an extension of the Sparse Roadside-Vehicle Communications approach. There is no change in infrastructure but the effective coverage of the network is increased by allowing vehicle-to-vehicle communication. As seen in figure 5.5 the range of communication hotspots is increased by allowing vehicles to forward data through other vehicles to the hotspot and the hotspot can also pass data back through the other vehicles. In this way vehicles which were outside the range in the previous approach may now be in range if a single or multi hop route to a hotspot exists. In heavy traffic when vehicles are bumper to bumper it is extremely likely that such a route exists. The hybrid approach can help overcome some of the challenges associated with the Sparse Roadside-Vehicle Communications by increasing the connection window in certain situations. This means that while a hybrid approach does not offer the always on connectivity of Ubiquitous Roadside-Vehicle Communications it offers an increase in connectivity over Sparse Roadside-Vehicle Communications given the same density of hotspots. These benefits do however come at the cost of a substantial increase in the complexity of the network. Technologies applicable for Sparse Roadside-Vehicle Communications can also be used for Hybrid Vehicle Communications however support for the formation of ad-hoc networks is an important consideration to allow vehicles to associate with one another for inter-communication.

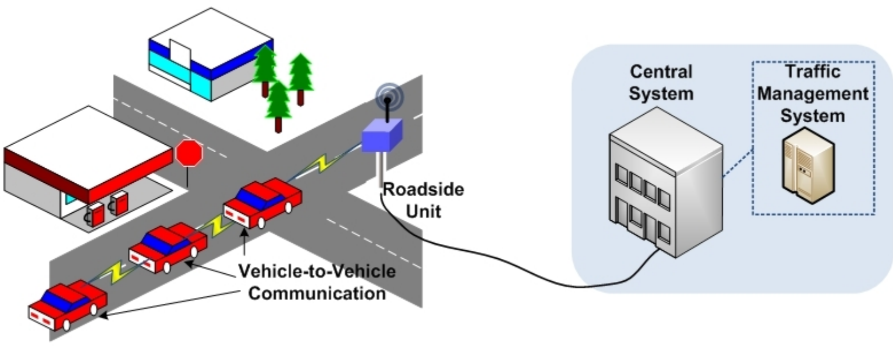


Figure 5.5: Inter-vehicle communication extends range of roadside unit

5.2 Sparse Roadside-Vehicle Communications Architecture

Ubiquitous roadside-vehicle communications would provide vehicles with always on connectivity anywhere on the road network. This would make it possible to provide a large array of services to vehicles. Unfortunately the large scale infrastructure necessary to provide an entire road network with high speed communication would require colossal investment. However many of the services, which can be provided with ubiquitous roadside-vehicle communications, can also be provided by sparse roadside-vehicle communications. In this

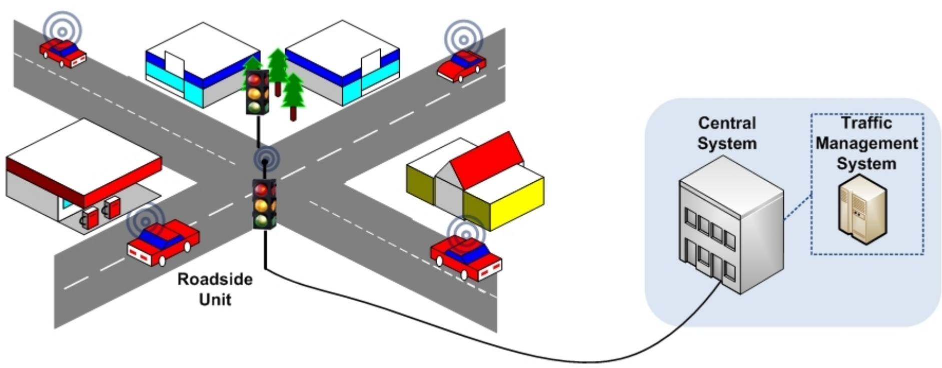


Figure 5.6: Roadside unit combined with existing infrastructure - traffic lights

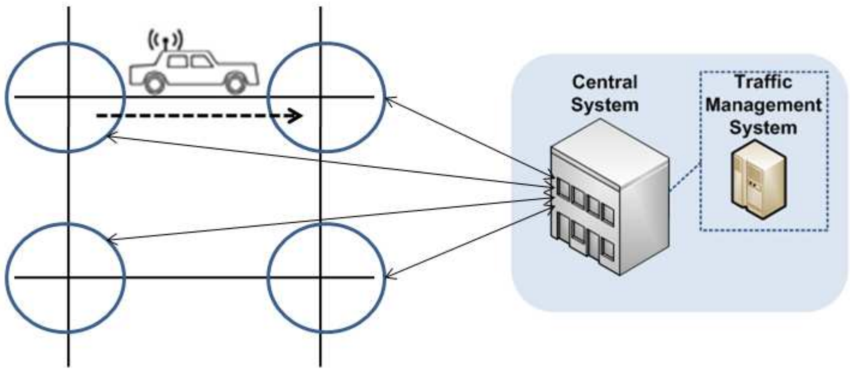


Figure 5.7: Vehicle moving through a sparse roadside-vehicle communications network

case vehicles communicate with intermittently available roadside infrastructure. We will discuss such an implementation in this and the remaining section of this chapter.

When deploying roadside units for sparse roadside-vehicle communications for convenience the infrastructure could be integrated with existing roadside infrastructure such as traffic lights (as shown in figure 5.6), which are present at every major intersection in built up areas. In figure 5.7 it is shown how integrated services can be provided by providing connectivity to a central system with communication hotspots in this example the services being provided are traffic management services. A vehicle is shown travelling in a Manhattan grid where the sparse roadside-vehicle communications network provides connectivity at each intersection. In this way a sparse roadside-vehicle communication system can be rolled out gradually while still providing benefits in the early stages of deployment.

In figure 5.8 the communication architecture for the sparse roadside-vehicle communication system is presented. The architecture is based on the European ITS communication architecture [97, 18]. That architecture is for a hybrid-vehicle communications system where there is inter-communication between vehicles as well communication between vehicle and roadside infrastructure. The architecture shown in 5.8 is a modified version of the European ITS communication architecture to reflect the transition from hybrid-vehicle communications to sparse roadside-vehicle communications.

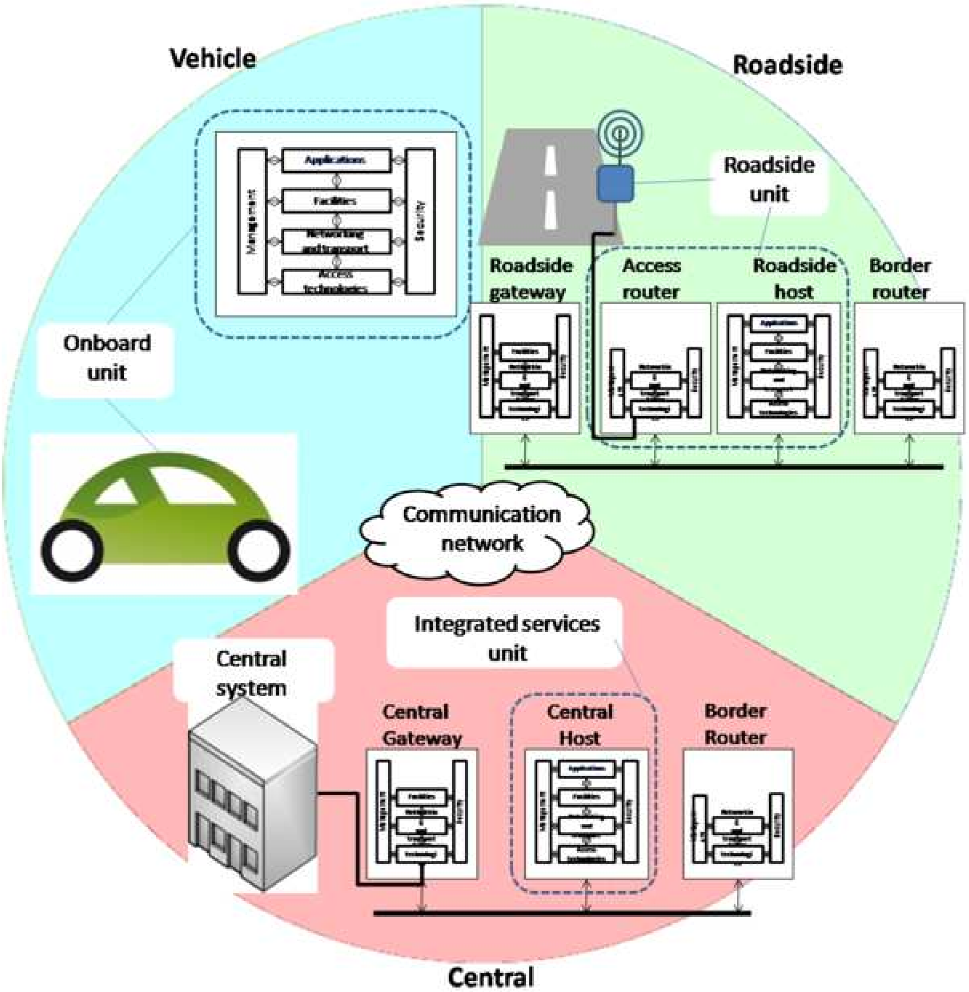


Figure 5.8: Sparse roadside-vehicles communications architecture

In terms of communicating entities this architecture comprises of three main entities vehicles, roadside units, and a central station. In the vehicle there is an on-board unit

- a device with a CPU, appropriate sensors such as GPS for location detection, a wireless transceiver to allow communications with roadside units and an interface that allows human interaction. This device can provide services such as traffic management services in the vehicle by communicating with roadside units. Examples of devices which could fulfil this role include smart phones, satellite navigation devices or a custom made thin client.

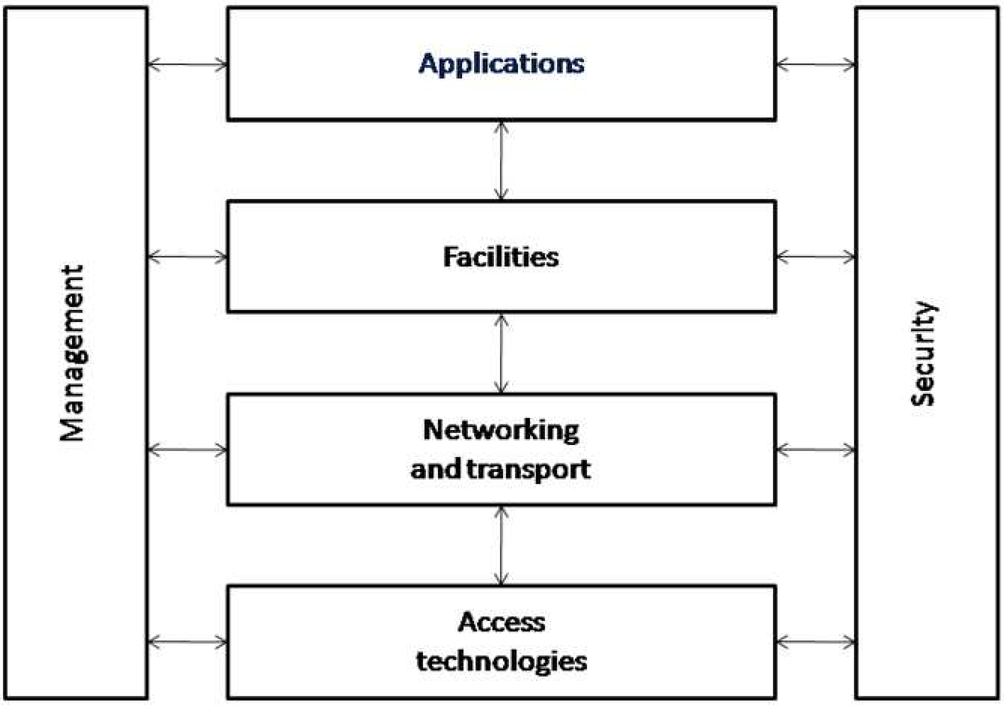


Figure 5.9: Communication protocol architecture

The roadside unit essentially provides communication with the central system in order to provide integrated services to vehicles. However depending on the use case it can also provide standalone services e.g. a roadside unit positioned at an e-tolling booth may provide an e-tolling service locally as well as allowing communication with the central system. The central system provides integrated services to vehicles via the roadside units. Traffic management services as described in the previous chapter are an example of services that require central management. The gateways included in the roadside and central entities as shown in figure 5.8, are present in order to allow connection to legacy systems if necessary in a specific deployment scenario.

From a communications perspective the vehicle, roadside and central equipment is based on the European ITS communication reference protocol stack [97] as shown in figure 5.9. This protocol stack is made up of four horizontal layers; access technologies, networking and transport, facilities and applications. These are flanked by a management and security layer. For the most part this is a relatively standard protocol stack but the facilities layer is specific to the vehicular environment. Services provided via vehicular communications will likely have many common requirements e.g. access to accurate location information. To facilitate rapid deployment the facilities layer provides access to standardized information, data and common functionalities.

5.3 Communication Principle

A vehicle travelling in a sparse roadside-vehicle communications network is not always connected. The vehicle's on-board unit provides connected services by receiving service updates whenever the vehicle passes through a communications hotspot provided by a roadside unit. Integrated services are provided by a central system via the roadside units. The typical communication sequence that occurs when a vehicle running an integrated service comes into the communication range of a roadside unit is shown in figure 5.10.

When a vehicle comes into range of a roadside unit it receives a notification advertising the existence of the unit (this is being broadcast periodically). Having received notification the vehicle's on-board unit then sends a service request to the roadside unit. For an integrated service the roadside unit forwards the request to the central system. The central system returns a service response to the roadside unit with the necessary information to update the service. This service response is forwarded to the vehicle's on-board unit by the roadside unit. On full receipt of the service update the on-board unit acknowledges the successful receipt of the service update ending the communication sequence between the

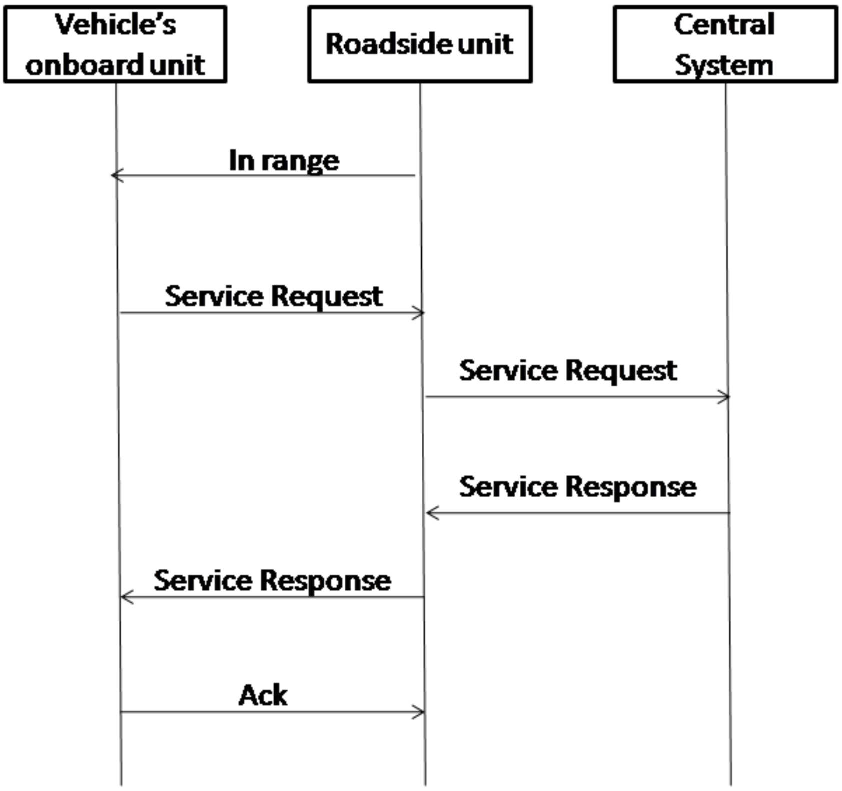


Figure 5.10: Typical communication sequence for sparse roadside-vehicle communications system

three entities.

A number of wireless technologies could be used in the implementation Wi-Fi, Bluetooth, UWB and ZigBee. There are however two existing technologies designed specifically for application in vehicular settings, DSRC 915 MHz and IEEE 802.11p Wireless Access in Vehicular Environments (A flavour of Wi-Fi designed specifically to work in the vehicular environment). IEEE 802.11p operates in the 5.9 GHz frequency band, which is also called the DSRC band and which should not be confused with the DSRC technology, which operates at the 915 MHz frequency band. These technologies were designed to meet the challenges of the vehicular environment such as communicating while travelling at speed and the likelihood of a short communication windows.

The DSRC standard has 12 megahertz of spectrum available in the 915 MHz frequency range and uses a single unlicensed channel. DSRC supports vehicle-roadside communication with a data rate of 0.5 Mbps and an effective range of thirty metres. DSRC was designed to support electronic toll collection but can also support other applications. IEEE 802.11p has 75 megahertz of spectrum available in the 5.9 GHz frequency range and has seven channels available. IEEE 802.11p supports data rates of 6-27 Mbps and has an effective range of up to one thousand metres. The technology was designed to support general internet access as well as traditional vehicle applications such as electronic toll collection, it supports both vehicle to roadside and vehicle to vehicle communications.

IEEE 802.11p is clearly a more applicable technology for this system. Take the example of a vehicle passing a roadside unit at 100km/h. DSRC has a range of 30 metres which gives a communication window of approximately 1.08 seconds. Given a data rate of 0.5 Mbps this gives a maximum data exchange of 0.54Mb. IEEE 802.11p has a maximum range of up to 1000m making a communication window of up to 36 seconds and with data rates of 6-27 Mbps this makes for maximum data exchanges in the range of 216 to 972Mb when a vehicle passes a roadside unit. It is clear that IEEE 802.11p allows greater amounts of data to be exchanged between vehicle's and roadside units. IEEE 802.11p was also designed with internet access in mind, which makes the job of connecting to a central system easier. The availability of additional channels is also beneficial as it can allow roadside units to overlap each others ranges without interference by operating on orthogonal channels. The support for vehicle to vehicle communication also adds flexibility as a sparse roadside-vehicle communications system could be deployed and be upgraded to a hybrid vehicle communications system at a later date if desirable.

If 802.11p is used then communication between the vehicle, the roadside unit and the central system occurs in the following way. In IEEE 802.11p roadside units transmit an on demand beacon 10 times per second, this announces to roadside units what applications it supports and on what channels. This beacon also contains all the information on-board units need to configure itself as a member of the roadside units Basic Service Set. An infrastructure basic service set is a group of IEEE 802.11 stations readied for communication by

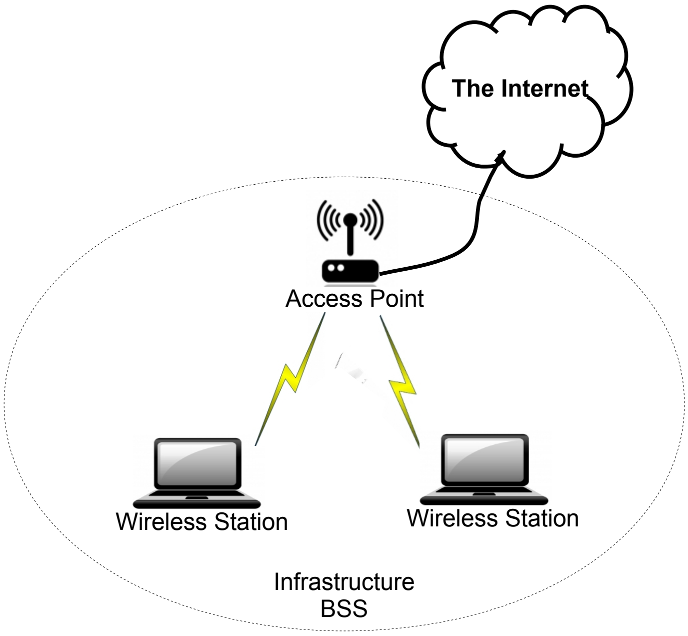


Figure 5.11: Infrastructure Basic Service Set

association with an access point as shown in figure 5.11. The Wi-Fi used in home networks or for public hotspots typically takes many seconds to set up a connection as there are a number of steps involved in joining the basic service set. Using IEEE 802.11p a vehicle's on-board unit can connect to a roadside unit by simply receiving the advertisement this only takes a number of milliseconds.

Once a vehicle has received an advertisement and joined a roadside units basic service set it is ready to communicate with the central system. As mentioned previously 802.11p has seven available communication channels. One of these is reserved as a control channel and two specifically for public safety applications consequently communication with the central system via the roadside unit takes place over one of the four remaining channels that are available for additional services as shown in figure 5.12. The roadside unit has a backhaul link to the internet and (this link may be wired or use a comparable wireless technology) acts as a gateway to the internet. The centralised system makes it services available over the internet using internet servers. In this way the vehicle is able to provide information to / receive information from the central system. The maximum amount of information that can be exchanged depends on how much time the vehicle spends in range of the roadside unit and how many other vehicles are in contention during this time.

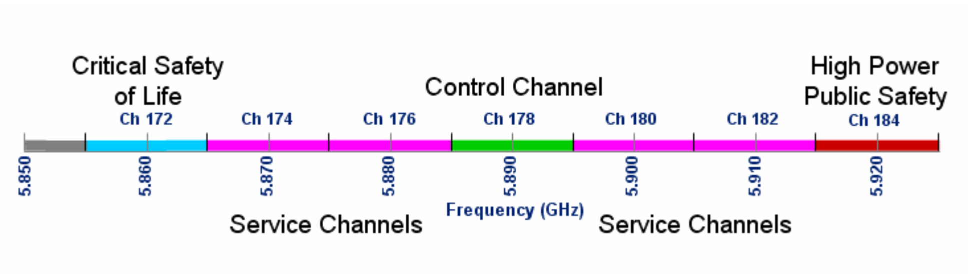


Figure 5.12: Channels available to IEEE 802.11p in the 5.9 GHz band [88]

5.4 Prioritisation of Services

The need for prioritisation of the services provided in a vehicle communications based system depends on the deployment scenario. The services which are likely to be provided to vehicles via vehicular communications are typically classified as:

- Traffic safety
- Traffic efficiency
- Value-added services (e.g., infotainment, business applications)

In a deployment scenario where all three of these service types are being provided it would be necessary to prioritize services with; traffic safety services receiving the highest priority, traffic efficiency services receiving the next highest priority and value-added services receiving the lowest priority.

In the infrastructure based sparse roadside-vehicle communication system described in this chapter services are provided by a central system. In this instance in a deployment scenario where traffic safety, traffic efficiency and value-added services are present prioritization may be provided at the application layer. An example of how this may occur is outlined in the communication sequence diagram in figure 5.13.

When a vehicle comes into range of a roadside unit it receives a notification advertising the existence of the unit. Having received notification the vehicle's on-board unit then

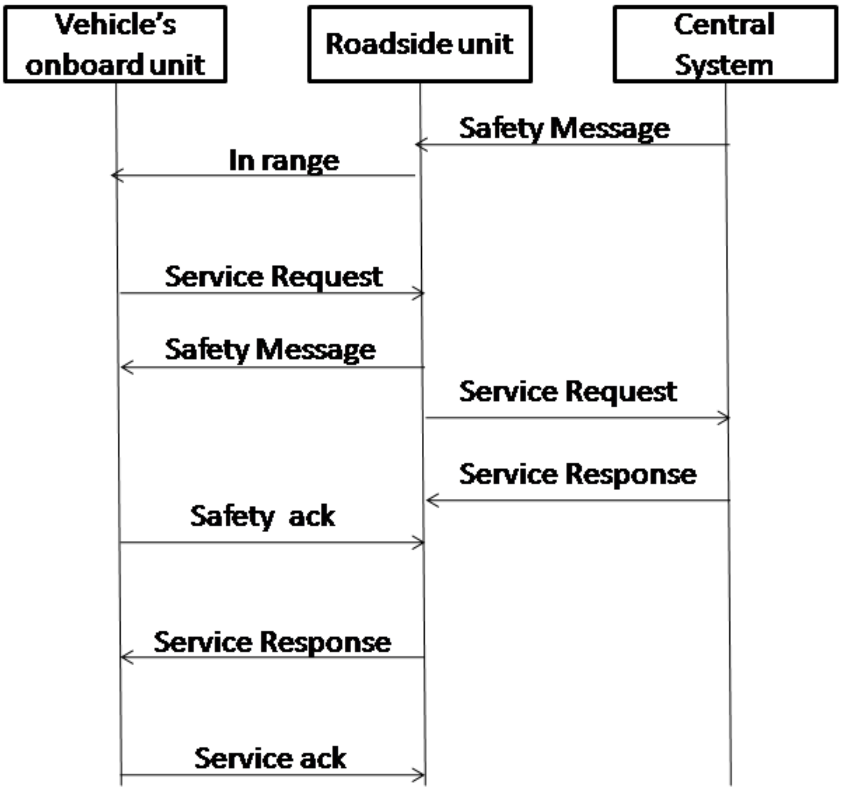


Figure 5.13: Typical communication sequence for sparse roadside-vehicle communications system

sends a service request to the roadside unit. However prior to the vehicle coming into range a traffic safety service running on the central system forwarded a safety message to the roadside unit. This message is relevant for all vehicles passing that roadside unit. Because safety services have highest priority the roadside unit forwards the safety message to the vehicle before forwarding the service request to the central system. The central system returns a service response to the roadside unit with the necessary information to update the service. When the roadside unit receives an acknowledgement from the vehicle's on-board unit indicating the safety message was received the service response is forwarded. On full receipt of the service update the on-board unit acknowledges the successful receipt of the service update ending the communication sequence between the three entities.

If the IEEE 802.11p communication technology is used then some provisions already exist for prioritisation of services. safety applications are prioritised over less critical services by 802.11p on-board unit. Roadside units advertise to on-board units ten times a second what applications it supports, on which channels. On-board units listen on the control channel, authenticate the roadside units digital signature, execute safety applications first, then switch channels and execute non safety application. When all applications have been executed the on-board units returns to listening on the control channel. It is important that a roadside unit also executes its non-safety applications such as traffic management applications in order of priority as there may not be sufficient time for all applications to communicate at every roadside unit passed. In practice an algorithm is required on the vehicle side to manage the servicing of non-safety applications communication requirements in order of priority. The concept is illustrated in figure 5.14. A vehicle's on-board unit is shown running a number of connected services. The communications these services require with roadside infrastructure points are shown being managed by a service communication manager, which is powered by a prioritized data exchange algorithm.

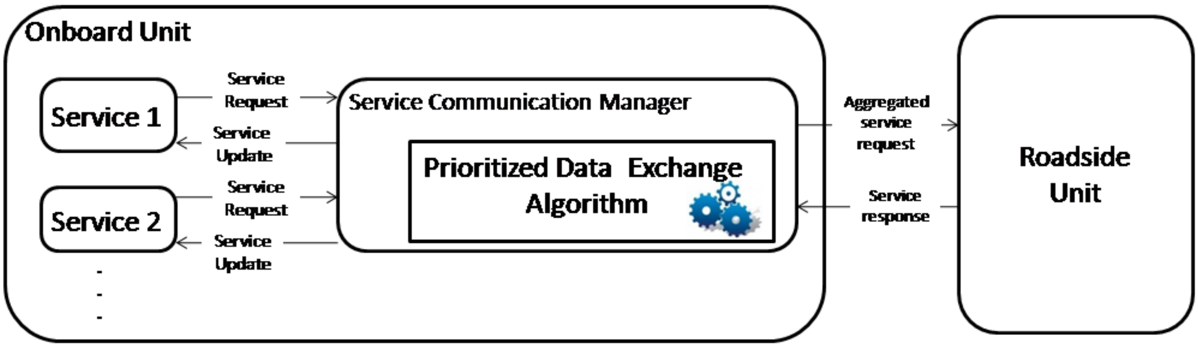


Figure 5.14: Service communication manager

The prioritized data exchange algorithm is outlined in the next chapter. The algorithm initiates and manages the exchange of data between a vehicle and roadside units it passes when travelling through a sparse roadside-vehicle communications network.

5.5 Chapter Summary

In this chapter an architecture for a sparse roadside-vehicle communications system for the provision of integrated services was introduced. After introducing the architecture the typical sequence of communication in a sparse roadside-vehicle communications system was explained and finally the need for prioritisation of services in such a system was discussed.

Chapter 6

Algorithms

The algorithms related to this thesis are detailed in full in this chapter. In chapter 4 a route management service enabled by vehicle communication was described. The best route selection algorithm which powers this service is described in section 6.1. In chapter 5 a service communications manager for a vehicle's on-board unit is described. The on-board unit runs connected services using vehicle communications and the service communication manager initiates and manages the exchange of information with the roadside units which provide connectivity. The service communications manager is powered by the prioritized data exchange algorithm described in section 6.3

6.1 Best Route Selection Algorithm Overview

In this section we will outline the main ideas behind the best route selection algorithm and explain the underlying mathematics. This is intended to give an understanding of how the algorithm works in principle. In the following section we will discuss practical matters regarding the implementation of the algorithm which should provide sufficient information to reproduce it. The Best Route Selection Algorithm (summarised in figure 6.1) is employed by the route management service in its decision making process. The best route selection algorithm is used to decide the best route for vehicles to take for a given journey. The best route decision is based on journey time, road capacity, fuel consumption and emissions.

A good route choice minimizes journey time, fuel consumption and emissions while also minimizing the vehicle's effect on traffic congestion. A new decision making process starts when a vehicle begins a journey by sending its origin and desired destination to the server. The steps listed next are followed:

1. Retrieve the k shortest routes from origin to destination. This is done by querying a cache of k-shortest paths using the origin and destination. It is possible to cache the paths as road layout changes infrequently relative to traffic conditions.

2. A fitness function is evaluated for each route, resulting in an associated fitness score. This fitness function which considers overall road congestion, vehicle journey times and fuel consumption is presented in the next section (eq. 6.5).

3. The best route is selected based on the fitness scores. The fitness function is the sum of weighted cost functions so the route with the lowest overall score is selected.

4. The user is given an instruction on what to do at the next junction to follow the chosen route. After passing through a junction and onto a new road segment, the journey origin is updated to that position and the algorithm is repeated. In this way the route instructions remain valid even if the user does not obey all instructions, and the route may also be altered if changes in traffic conditions mean a better route now exists.

The cache of k shortest paths is created using the generalised Floyd Shortest-Path algorithm [54]. Constraint checking runs in parallel to ensure the k routes are valid i.e. no rules of the road are broken (e.g. going the wrong way down a one way street). The value of k is important as only that number of routes will be considered as possible solutions. The parameter was introduced to reduce the complexity of the solution space and speed up the solution finding process.

Other well known k shortest path algorithms exist including the double sweep algorithm and the generalized Dantzig algorithm [54]. However the double sweep algorithm is better suited to finding the k shortest paths between a specified vertex and all other vertices, while

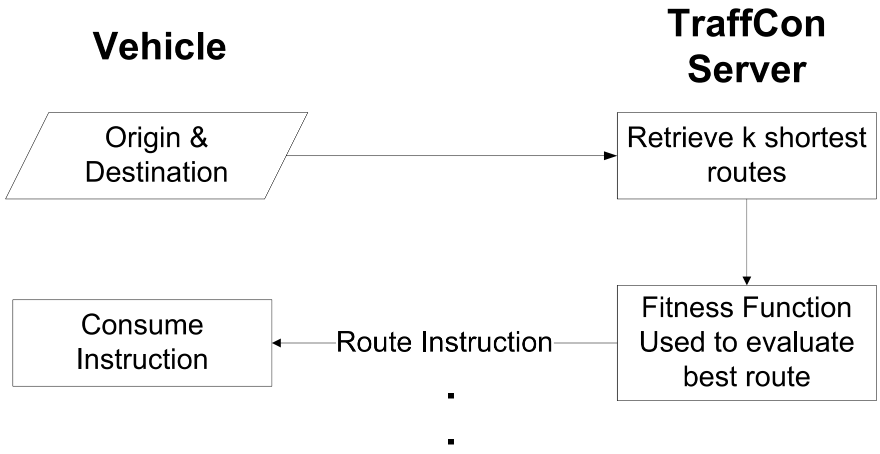


Figure 6.1: One Iteration of TraffCon's Best Route Selection Algorithm

both Floyd and Dantzig algorithms are suited to finding the k shortest paths between every pair of vertices, as required. These two algorithms are of the same order of complexity so the choice for the Floyd algorithm does not influence system performance.The complexity of the proposed algorithm is discussed further in the next sub section.

6.1.1 Complexity

In a network such as a wireless vehicular network where communication delays may be substantial it is important that delays introduced into any communication sequence by processing are kept to an absolute minimum. It was for this reason that the k parameter was introduced.

The generalised Floyd Shortest-Path algorithm [54] is used to create the cache of paths off-line, from the current roadmap. Constraint checking runs in parallel to ensure the k routes are valid i.e. no rules of the road are broken (e.g. going the wrong way down a one way street). The value of k is important as only that number of routes will be considered as possible solutions by the algorithm. If a permanent change is made to the road layout (e.g. new road constructed) then the cache of paths is regenerated using the new roadmap. Temporary changes in the road layout can be accounted for by associating special fitness scores to a road segment e.g. for a road closed due to roadworks the road segment can be assigned a fitness score of infinity so it is never selected.

In terms of computational complexity the algorithm is the simple operation of choosing the smallest number from a list of size k meaning a complexity of order $O(k)$. In practice k will be a small number i.e. approximately less than or equal to 20 so processing time will be negligible.

An obvious alternative to the proposed approach is to use a shortest path algorithm with the fitness scores as edge weights. However this results in a complexity of order $O(V^2)$ where V is the number of vertices in the road network. In a large-scale real-world road network, V is a very large number and consequently the proposed solution will be considerably more efficient.

6.1.2 Preliminaries: Quantifying the Effect of Route on Fuel Consumption and Emissions

In this section an equation which quantifies the effect of route on fuel consumption and gas emissions is derived. It is used as part of the fitness function in the next subsection.

Equation (6.1) can be used to estimate the value of the fuel consumed (ml), ΔF, during a time interval of duration Δt seconds [5].

$$fuel.\Delta F = [\alpha + \beta_1 R_T v + (\beta_2 M_v a^2 v/1000)_{a>0}]\Delta t \tag{6.1}$$

where,

R_T = total tractive force (kN) required to drive the vehicle, which is the sum of the rolling resistance, air drag force, inertia force and grade force, is given by equation (6.2)

M_v = vehicle mass (kg)including occupants and any other load,

v = instantaneous speed (m/s) = v (km/h) / 3.6,

a = instantaneous acceleration rate (m/s^2), negative for deceleration,

α = constant idle fuel rate (mL/s), which applies

during all modes of driving (as an estimate of fuel
used to maintain engine operation),

β_1 = the efficiency parameter which relates fuel
consumed to the energy provided by the engine, i.e.
fuel consumption per unit energy (mL/kJ), and

β_2 = the efficiency parameter which relates fuel
consumed during positive acceleration to the product
of inertia energy and acceleration, i.e. fuel
consumption per unit of energy-acceleration (ml/(kJ.m/s^2))

$$\Delta R_t = b_1 + b_2 v^2 + g(Ma/1000)(G/100) \tag{6.2}$$

where,

b_1 = the tractive force parameter for rolling resistance,

b_2 = the tractive force parameter for aerodynamic resistance,

g = is gravitational acceleration (ms^{-2}) and,

G = is the percent gradient.

Carbon Monoxide (CO), Hydrocarbons (HC) and Nitrogen Oxides (NO_x) emissions are calculated similarly. The differences are; α = constant idle emissions rate (g/s), β_1 = the efficiency parameter which relates pollutant emitted to the energy provided by the engine, i.e. emission per unit energy (g/kJ), and β_2 = the efficiency parameter which relates pollutant emitted during positive acceleration to the product of inertia energy and acceleration, i.e. emission per unit of energy-acceleration (ml/(kJ.m/s^2)).

The values of Carbon Dioxide (CO_2) emission are estimated directly from fuel consumption:

$$\Delta F = f_{CO_2} \Delta F \tag{6.3}$$

where,

ΔF = fuel consumption in mL calculated from (6.1) and,

f_{CO_2} = CO_2 rate in grams per millilitre of fuel(g/ml).

When comparing the fuel consumed or emissions produced by the same vehicle along a number of alternative routes M$_v$, α, β_1, β_2, b_1, b_2 and f_{CO_2} remain constant for all routes. Therefore when evaluating the contribution of the route to fuel consumption typical values may be used for these constants. The parameters M$_v$, α, β_1, β_2,b_1 and b_2 are set to the typical values derived for light vehicles in [39] as shown in table 6.1.

Parameter	M_v	α	β_1	β_2	b_1	b_2
Value	1400 kg	3.75 L/s	9x10e -4 L/J	3x10e -4 L/J	.233 N	7.9x10e -4 kg/m

Table 6.1: Parameter values for the fuel cost equation

The following cost function equation (6.4) is used to compare routes when evaluating routes in terms of fuel consumption and emissions. Given the case where the route is simply a link The Fuel Consumption and Emissions Cost for a link n can be calculated as the summation of the instantaneous fuel consumption's ΔF for all the time intervals along the given link from its origin (O) to destination (D).

$$f_n = \sum_{j=O}^{D} \Delta F_j \tag{6.4}$$

This equation will be used as part of a larger function in the next section.

6.1.3 Fitness Function

The fitness function (6.5) is proposed to choose a vehicles route so that journey time, congestion and fuel consumption are minimized. Parameters for overall and individual benefit are used e.g. Journey Time Cost and Used Capacity Cost.

The fitness function consists of weighted cost components including T which encourages a routing solution with the minimum possible user journey time, C ensures the solution considers the effect on congestion, S was introduced to assure fairness i.e. prevent individuals being excessively rewarded/punished.

Each of the components are weighted by w_i (6.6), to force the emphasis on a particular outcome. The more important a component is considered to be to the solution, the smaller the weighting factor associated with it, and therefore the stronger its contribution to the overall score R_{nv}.

$$R_{nv} = w_1 T_{nv} + w_2 C_{nv} + w_3 F_{nv} \tag{6.5}$$

Given a certain vehicle v taking route n,

T_{nv}.............Journey Time Cost

C_{nv}.............Used Capacity Cost

F_{nv}.............Fuel Consumption and Emissions Cost

w_i.............Weighting Factors

$$\sum_{i=1}^{3} w_i = 1 \tag{6.6}$$

It is possible to enhance this fitness function at a later date by considering additional parameters e.g. speed, emissions (i.e. CO_2, CO, HC, NO_x), jitter, operating cost etc.

6.1.4 Individual Cost Components

The information made available by the data gathering and processing stage is pulled as required by the fitness function to evaluate its constituent cost functions. Each cost function generates a cost score which is expressed as a percentage. These individual cost scores are calculated as follows.

6.1.4.1 Journey Time Cost

The journey time cost for a vehicle v taking route n is calculated as the summation of link times (t) from origin (O) to destination (D) along the given route over the maximum journey time from the K possible routes t_{maxv} (6.7).

$$T_{nv} = \sum_{j=O}^{D} t_j / t_{maxv} \tag{6.7}$$

This function encourages the fastest route (in temporal terms) to be selected.

6.1.4.2 Used Capacity Cost

The used capacity cost for a vehicle v taking route n is calculated as the average of the segment capacity (l) adjusted used capacities (c) of all the segments () from origin (O) to destination (D) along the given route over the maximum average c_{maxv}. N is the number of segments along the route (see equation 6.8).

$$C_{nv} = \sum_{j=O}^{D} (c_j * l_j / N) / c_{maxv} \tag{6.8}$$

This function encourages the least congested route to be selected.

6.1.4.3 Fuel Consumption and Emissions Cost

The Fuel Consumption and Emissions Cost for a vehicle v taking route n can be calculated as the summation of the individual link fuel consumption and emissions costs(f (6.4)), for all the links along the given route from origin (O) to destination (D) over the maximum fuel consumption and emissions Cost from the K possible routes f_{maxv} (6.9).

$$F_{nv} = \sum_{j=O}^{D} f_j / f_{maxv} \tag{6.9}$$

This function encourages the route which will result in the least amount of fuel being consumed and in the least amount of emissions being emitted, to be selected.

6.2 Best Route Selection Algorithm Implementation

When implementing the best route selection algorithm there are a number of practical considerations which must be made. In order to illustrate how the algorithm may be implemented three main aspects will be examined; mapping and location, network considerations, and data aging.

6.2.1 Mapping and location

In order to calculate routes for vehicles it is necessary to have a detailed digital map of the area that they are travelling in. This map must describe the road network in full including important information that affects the ability to travel along a given route such as; the number of lanes on a street and their directionality, what streets a vehicle may enter if travelling in a certain direction on a street (In cities right or left turns off a street may often be prohibited as seen in figure 6.2), if a street is one way etc. A geographical information system captures, stores, analyses, manages, and presents data that are linked to location(s). The digital map serves as the base for any geographical information system and additional information is overlaid on the digital map base. For example one layer of information may relate to traffic conditions and this traffic may be associated with sections of road as described in the map.

Once a base map and a geographical information system to manage information related to that map are in place add-ons can be built for the system which perform operations on the data to generate further information for services. In order to implement a traffic management service based on the best route selection algorithm for a geographical area the following steps must be taken.

- Select a suitable Geographical Information System and Digital Map Base
- Build an add-on to calculate the cache of k-shortest paths for the road network
- Create suitable database tables in the Geographical Information System to store the location based traffic data for the fitness function

Figure 6.2: The route a vehicle may take is often limited by traffic restrictions[1] [2]

In practice for this thesis when implementing the best route selection algorithm for simulation based testing the OpenJUMP Geographical Information System [3] was used with base maps taken from the US Census Bureau's TIGER/Line file on-line database[4]. The TIGER/Line files are spatial extracts from the Census Bureau's MAF/TIGER (Master Address File/Topologically Integrated Geographic Encoding and Referencing) database, containing features such as roads, rail roads, rivers, as well as legal and statistical geographic areas, these files are available free of charge. The OpenJUMP system is easily extensible by building Java based add-ons. An add-on which implements the generalized Floyd shortest-path algorithm was built to calculate a cache of the k-shortest paths for the road network layer of data found in the TIGER/Line files as seen in figure 6.3. This allowed the cache needed for the best route selection algorithm to be generated for any subset of the US road network, more information on how the k-shortest paths were calculated can be found in Appendix A.

Once a map base is established for the geographical information system it can be assumed that vehicles will use the same map in their on-board units and will have GPS there-

[1]One way sign Crossgar, 2010, WikiMedia Commons, Northern Ireland, viewed 31 March 2010, http://commons.wikimedia.org/wiki/File:One_way_sign,_Crossgar,_February_2010.JPG

[2]No Left Turn Sign, 2007, WikiMedia Commons, France, viewed 31 March 2010, http://commons.wikimedia.org/wiki/File:Photo_263.jpg

[3]http://www.openjump.org

[4]http://www.census.gov/geo/www/tiger/

fore, they can tell where on the map they are. This means any data collected by sensors on-board the vehicle can be geographically referenced. In order for the system to gather information from vehicles and for vehicles to receive instructions some network infrastructure must be in place to allow communication. The effect the available network infrastructure has on the implementation of the best route selection algorithm is covered in the next section.

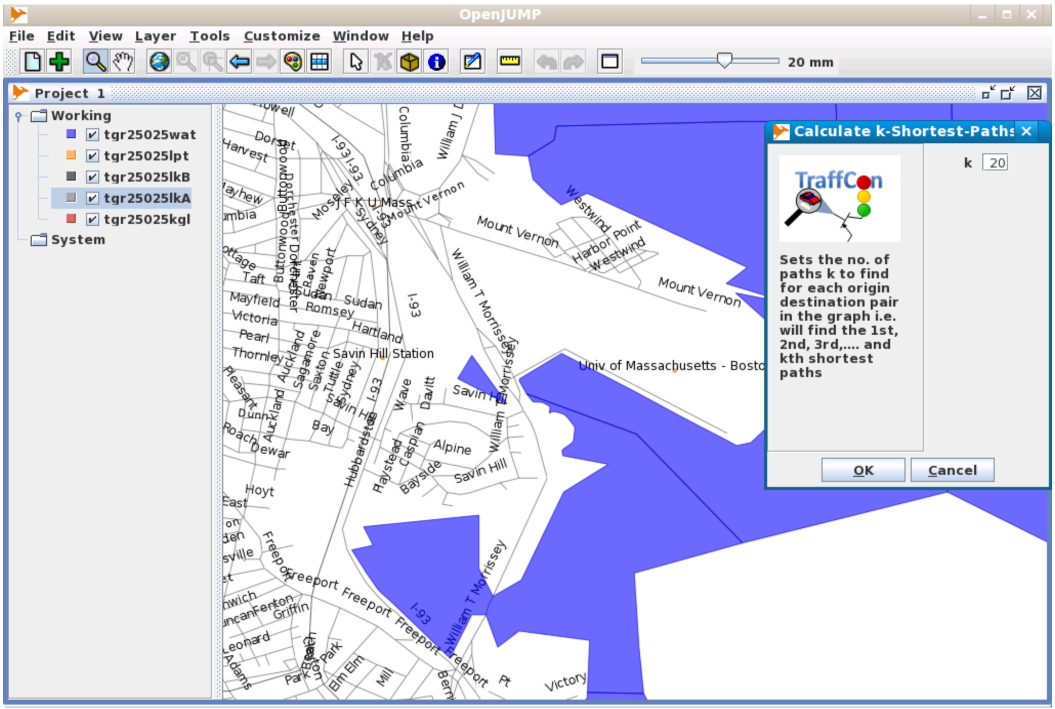


Figure 6.3: OpenJUMP runs the k-shortest paths add-on with a map of Boston, MA, USA

6.2.2 Network Infrastructure

Given a centralised approach there must be some level of network infrastructure in order to support communication between vehicles and the central system. The nature of this network infrastructure will affect the implementation of the best route selection. It may also be possible to implement the best route selection algorithm using a fully distributed approach the algorithm would run locally on the on-board unit of each vehicle and its effectiveness would depend on the ability of vehicles to disseminate traffic information amongst themselves using ad-hoc communication. The distributed case is however considered to be beyond the scope of this work. Taking the centralised case the infrastructural solutions may be divided into three basic approaches:

- Ubiquitous Roadside-Vehicle Communications
- Sparse Roadside-Vehicle Communications
- Hybrid Vehicle Communications

For ubiquitous roadside-vehicle communications the network infrastructure places little or no limitations on the implementation of the algorithm. This approach provides full coverage giving always on connectivity which allows vehicles to contact the server any time, any where. This means vehicles can send data to the server whenever the want and request instructions whenever it needs.

However for both sparse roadside-vehicle communications and hybrid vehicle communications the vehicle is only capable of contacting the server when close to a communication hotspot. The number of communication hotspots can dramatically influence the operation of the algorithm. For example if the density of hotspots deployed is too low then the server may not be able to gather enough information on traffic conditions for the algorithm to be effective and vehicles may not be able to receive instructions regularly enough to route them to their destination. In a worst case it may be possible for vehicles to complete a journey without ever passing a hotspot making use of the best route selection algorithm powered traffic management service impossible for some journeys. In practice for this thesis when implementing the best route selection algorithm for simulation based testing a sparse roadside-vehicle communications network was used. The density of hotspots for this network was one communications hotspot placed at every intersection in the road network. Data gathering and related issues such as data aging are discussed in the next section.

6.2.3 Data Gathering

In the best route selection algorithm a fitness function is evaluated for potential routes, resulting in an associated fitness score. The route with the best fitness score is chosen. This fitness function considers overall road congestion, vehicle journey times and fuel consumption. The fitness score is a sum of the associated costs for congestion, journey time and fuel consumption, the best score is the lowest of those compared. In practice in order to implement the best route selection algorithm data about the state of the road network must be gathered in order to calculate costs associated with routes.

In practice for this thesis when implementing the best route selection algorithm for simulation based testing the vehicles travelling in the road network gather this data. Measurements are associated with road segments; a road segment is a section of road between two junctions and can be seen as analogous to an edge in graph theory. A route is made up of one or more segments and the measurements for each road segment along that route are combined to calculate the cost associated with that route. Vehicles have an on-board unit which has a map of the road network and GPS so they know where they are on that map. This means vehicles can tell when they enter/leave a road segment and so can take measurements while on that road segment. These measurements are sent to the server using the access point found at the junction when leaving the road segment. The server keeps a database of these segment related measurements indexed by a unique identifier for each segment. The measurement taken by vehicles while traversing a road segment are used to calculate

- Journey Time Cost
- Used Capacity Cost
- Fuel Consumption and Emissions Cost

When vehicles send measurements related to journey time, used capacity and fuel consumption at the end of a segment the data is prefixed with a unique identifier for the vehicle and a time-stamp. In order to calculate journey time cost for a route the time (t) taken to cross each segment must be known. To make this information available a simple journey time measurement is taken, a timer is set when the vehicle enters the segment and is stopped when the vehicle leaves. At the end of a segment the id of the segment and the time taken to cross it is sent the time value associated with the segment is updated using a windowed average. With this information available the server may calculate the journey time cost of a route using equation (6.7) whenever necessary.

In order to calculate used capacity cost for a route the used capacity (c) of each segment must be known. Used capacity is the length of road occupied by vehicles divided by the available road length on a segment. In order to measure used capacity the length of vehicles on a segment must be known so a count is kept of the length of vehicles on each segment. If the vehicle is leaving a segment it must be entering a new segment, the ID of that segment is sent along with the data for journey time cost which contained the ID of the last segment. This allows the count for the length of vehicles on the new segment to be increased and the count for the last to be decreased. A database trigger can update the used capacity value of a segment when the count for the length of vehicles on a segment changes. With this information available the server may calculate the used capacity cost of a route using equation (6.8) whenever it needs to.

In order to calculate fuel consumption and emissions cost for a route the fuel consumption (f) of each segment must be known. To calculate the fuel consumption value for a segment, vehicles must sample their velocity and acceleration at a suitable rate while crossing the segment. The instantaneous values are used in equation (6.1) to calculate instantaneous fuel consumptions and these values are summed according to equation (6.4) to generate the total fuel consumption while on that segment. That fuel consumption value is sent to the server along with the data associated with journey time cost and used capacity cost, the fuel value associated with the segment is updated using a windowed average. With this information available the server may calculate the fuel consumption and emissions cost of a route using equation (6.9) whenever it is required.

In summary vehicles gather data for the traffic management server, this data is sent using the access point found at the junction when leaving a road segment, the data sent allows the calculation of the Journey Time Cost, Used Capacity Cost and Fuel Consumption and Emissions Cost of routes for use in the best route selection algorithm. The data sent is associated with individual road segments and can be summarised as follows (or as seen in table 6.2), id of vehicle sending data, time-stamp, id of last segment, time taken to traverse that segment, id next of next segment and fuel cost.

Data Field	Value	Note
Vehicle ID	01 C 9479	A unique identifier for the vehicle in this example the registration number is used
Time Stamp	2009-05-11 07:57:46	The time the data was sent
Last Segment ID	-71.040778+42.285664-71.040694+42.285698	A unique identifier for the segment the vehicle is leaving. In this case the GPS co-ordinates of the junctions which bookend the road segment form its id
Segment Time (ms)	13000	The time taken to traverse the last road segment
Next Segment ID	-71.040694+42.285698-71.040584+42.285741	A unique identifier for the road segment the vehicle is entering
Fuel Consumption (ml)	4	The level of fuel consumption associated with crossing the last road segment

Table 6.2: Summary of data sent by vehicles

An important concern for a system such as the traffic management system described is data aging i.e. instances where the available data becomes too old and is no longer relevant to the current situation. For this system barring a system failure there are two likely instances where data may not be updated for a significant period of time. The first is where there is very little traffic and there has been no new data gathered for some road segments for a considerable length of time. In this case data aging is not an issue as segments can not go from congested to empty instantaneously so the data may be old but it will reflect that the segment is free of traffic. The other is when traffic is very heavy and a vehicle is stuck in the centre of a segment for a significant period of time. In this case the data the vehicle is operating on may be come outdated however it cannot act on that information until it reaches the junction at the end of the segment at which point the information will be updated and any instructions given will be based on relevant information. In both cases the available data may age but at no point are decisions being made based on data which is not relevant. Issues related to data aging and the relevance of data used to make decisions are important considerations for this thesis, a more in depth discussion on data freshness and system latency is given in the next section.

6.2.4 Data Freshness and System Latency

In the previous section we saw how the data gathered was turned into the information used by the best route selection algorithm i.e. journey time cost, used capacity cost, and fuel consumption and emissions cost. In the case of journey time cost, and fuel consumption and emissions cost a windowed average was used. In practice for this thesis when implementing the best route selection algorithm for simulation based testing the window size was set at ten.

The purpose of this windowed average was to eliminate "noise" or unwanted data which may lead to an inaccurate description of the road network. Examples of unwanted data include; a vehicle which drives very slowly (significantly below the speed limit) on an empty street resulting in a journey time cost which gives the impression the street is more congested than it is or a vehicle which drives erratically on an empty street braking and accelerating more than necessary for conditions on the road resulting in a fuel consumption and emissions cost which is too high. If instantaneous values were used (a window size of one) the data would be as fresh as possible but the effect of "noise" would be at its highest. A larger window can eliminate noise, however if a very large window size is used then the freshness of the data is compromised. For example if a window size of 100 is used, a journey time cost calculated for a route uses information about road segments containing data from the hundredth last vehicle to leave the road segments. Depending on road network condition a considerable amount of time may have passed since the data from the hundredth vehicle was received. The window size of ten was chosen to find a balance between the need to eliminate unwanted data and the need to keep data as fresh as possible.

When taking the measurements needed to determine used capacity costs a windowed average was not used. This is because the system was measuring how many vehicles were on each road segment. A vehicle is either on a road segment or it is not the only thing affecting the freshness of this data is how quickly a vehicle informs the system when it changes road segment, with an access point at every intersection vehicles can inform the system as they change road segment.

In any system where measurements are taken and then decisions are made based on these measurements a certain latency is inherent as the measurements are no longer current by the time the decision is taken. For this system the state of the road network is captured by taking measurements related to road segments and then a route decision is taken. A very long route is made up of many road segments and by the time a vehicle has made it to one of the later road segments traffic conditions may have changed significantly. This is why as described in section 6.1 the algorithm adapts over the course of a journey, updating its origin to its current position and recalculating the best route from origin to destination every time a vehicle passes a communication hotspot. In this way the route may be altered if changes in traffic conditions mean a better route now exists. In practice it was found that a "flash crowd"[1] effect occurred if the route was adapted from the first segment of the remaining route, vehicles oscillate from segment to segment rather than completing their journey. In order to overcome this problem in the implementation vehicles adapt their journey from the second segment of the remaining route rather than the first. As stated previously for this thesis a sparse roadside-vehicle communications network was considered, with a density of hotspots of one communications hotspot placed at every intersection in the road network. In an implementation with lower hotspot density latency may become an issue as adaptation can occur less frequently. One way of overcoming this may be to use historic data to predict how traffic conditions may change.

6.3 Prioritised Data Exchange Algorithm Overview

Providing vehicles with communication capabilities opens up a range of application/services which were previously unavailable. The services which are likely to be provided to vehicles via vehicular communications are typically classified as [97]:

- Traffic safety

[1] This phrase "Flash Crowd" originates from the 1973 novella by science fiction author Larry Niven, one of a series about the social consequence of inventing an instantaneous, practically free transfer booth that could take one anywhere on Earth in milliseconds. One consequence not foreseen by the builders of the system was that with the almost instantaneous reporting of newsworthy events, tens of thousands of people worldwide along with criminals would flock to the scene of anything interesting, hoping to experience or exploit the instant, thus disorder and confusion became widespread.

- Traffic efficiency
- Value-added services (e.g., infotainment, business applications)

If these types of services are to be provided in this sparse roadside-vehicle communications environment they are number of issues which must be considered.

- Connectivity
- Connection Time
- Variable Infrastructure Density

In a sparse roadside-vehicle network vehicles are not always connected. Connectivity is provided intermittently at communication hotspots as seen in figure 6.4.A. If providing a location based service for example the vehicle may not be connected at the location where information for a service is required. Therefore there are times when information must be gathered before it is needed.

When a vehicle moves into range of a hotspot the connection window is often very short as its mobility can take it out of range quickly as shown in figure 6.4.B. For example take the case of a roadside unit in an urban area that has an effective range of 100m. If traffic is moving slowly and vehicles are passing at 10km/h then the connection window lasts approximately 36 seconds. If traffic is moving freely and vehicles pass by close to the speed limit - at 60km/h then the connection window only lasts approximately 6 seconds, a very short period of time in which to establish a connection and exchange information.In order for services to function the necessary information must be exchanged in an efficient manner while connected. The problem can be combated to some extent by infrastructure placement. Intersections make a good location for roadside units, vehicles slow down and/or stop at intersections potentially increasing the duration of the connection window. For example take a simplified scenario where a roadside unit with an effective range of 100m is placed at a signalised intersection, given a vehicle averages 10km/h when moving through the units range and a vehicle can spend up to 30 seconds stopped at the lights then the connection window can potentially exceed a minute and ranges from a duration of approximately 36 to

66 seconds. Placing roadside units where vehicles are likely to be travelling at slower speeds increases the connection window, it may also be increased by lengthening the effective range of roadside unit.

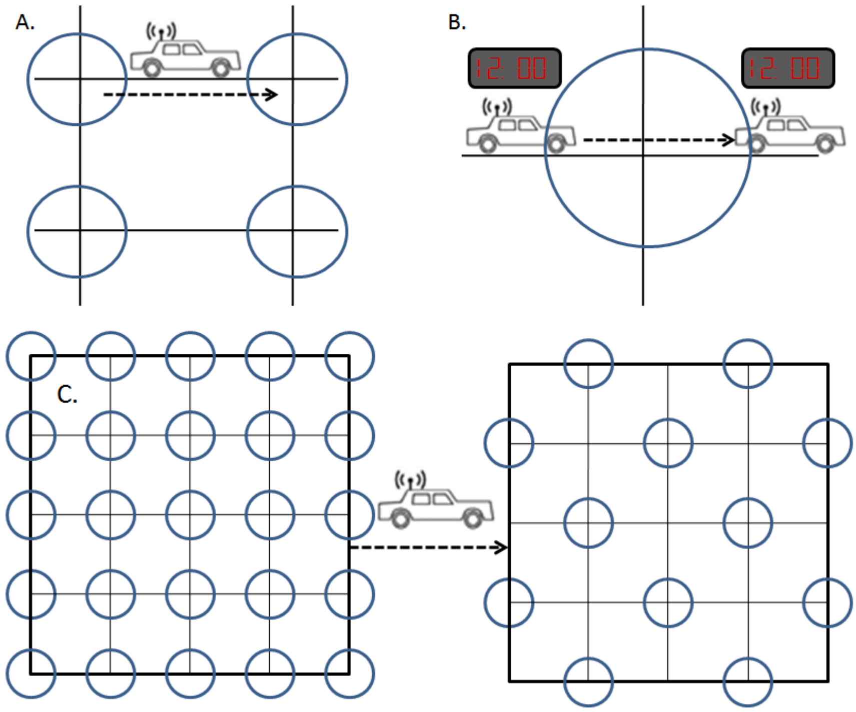


Figure 6.4: Connectivity, connection time and variant infrastructure density issues in a Manhattan grid.

In practice for this thesis a sparse roadside-vehicle communications network was considered, with a density of hotspots of one communications hotspot placed at every intersection in the road network. It is possible to have scenarios where there is a less uniform deployment of communication hotspots. In such cases the frequency with which a vehicle may encounter a roadside unit can vary widely depending on the infrastructure density in a given area. In figure 6.4.C we see an example where a vehicle is moving from a region with a communications hotspot at every intersection to an area where hotspots are found at half the intersections. Variations in the density of roadside infrastructure will affect the level of connectivity a vehicle receives which in turn affects the provision of services. Services should be capable of degrading and upgrading gracefully depending on the level of connectivity available.

It is because of these issues that prioritization is essential. When a connection is available services must be given access to the connection in order of priority so that at a minimum the most important services will continue to function. Vehicles will request services in order of priority and the infrastructure points will deliver the data for these services in order of priority so the exchange of data can be said to have been prioritized. In this way even if the amounts of data which can be exchanged fluctuates essential services may be maintained. In order to achieve the necessary prioritization of communication resources available to services the prioritized data exchange algorithm is proposed . As mentioned in chapter 5 and seen in figure 6.5 this algorithm is used by a service communication manager on the vehicles on-board unit.

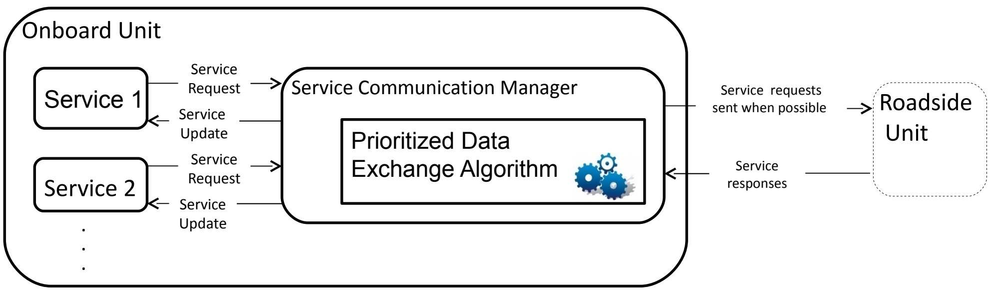


Figure 6.5: Service communication manager uses prioritised data exchange algorithm

While a vehicle is on a journey its on-board unit may be running any number of services which must talk to a server periodically, such as the traffic management service described in this thesis. The on-board unit has a service communication manager that receives service request messages from services that wish to contact the server. These are handled by the prioritized data exchange algorithm (summarised in figure 6.6) which operates according to the following steps:

1. Maintain queue of request messages from services in order of service priority while waiting for a connection to become available (messages for the service with highest priority will be at the head of the queue). If two services of equal priority send service request messages then they are ordered by first in, first out.

2. When connection becomes available, send messages in order of priority. Server processes messages and responds if necessary (messages may require a response take for example a message requesting new information such as a route recommendation but some may not such as a message supplying sensor data to the server).
3. When connection is dropped return failure to communicate messages to services which did not get through
4. Empty queue and repeat from step one.

This algorithm assumes that all services are assigned a priority. Services which have a high priority will likely get access to the network when it is available those with a low priority may not always get a chance to communicate at communication hotspots. The assignation of priority is an implementation issue and this and other such issues will be discussed in the next section.

6.4 Prioritized Data Exchange Algorithm Implementation

When implementing the prioritized data exchange algorithm there are a number of practical considerations which must be made. In order to illustrate how the algorithm may be implemented three important aspects will be examined; the on-board units specifications, the communications technology used and priority assignation.

6.4.1 On-board unit specifications

The hardware and system specifications of the on-board unit could potentially impact the implementation of the algorithm. Tight memory and processing constraints could limit the queue size. However given the space and constant power supply available in a vehicle and cost aside these should not be an issue. The communications hardware used will significantly affect the implementation and this is discussed in greater detail in the next section. Other system specifications such as the operating system used could also influence the implementation. There are a wide range of operating systems available from long established

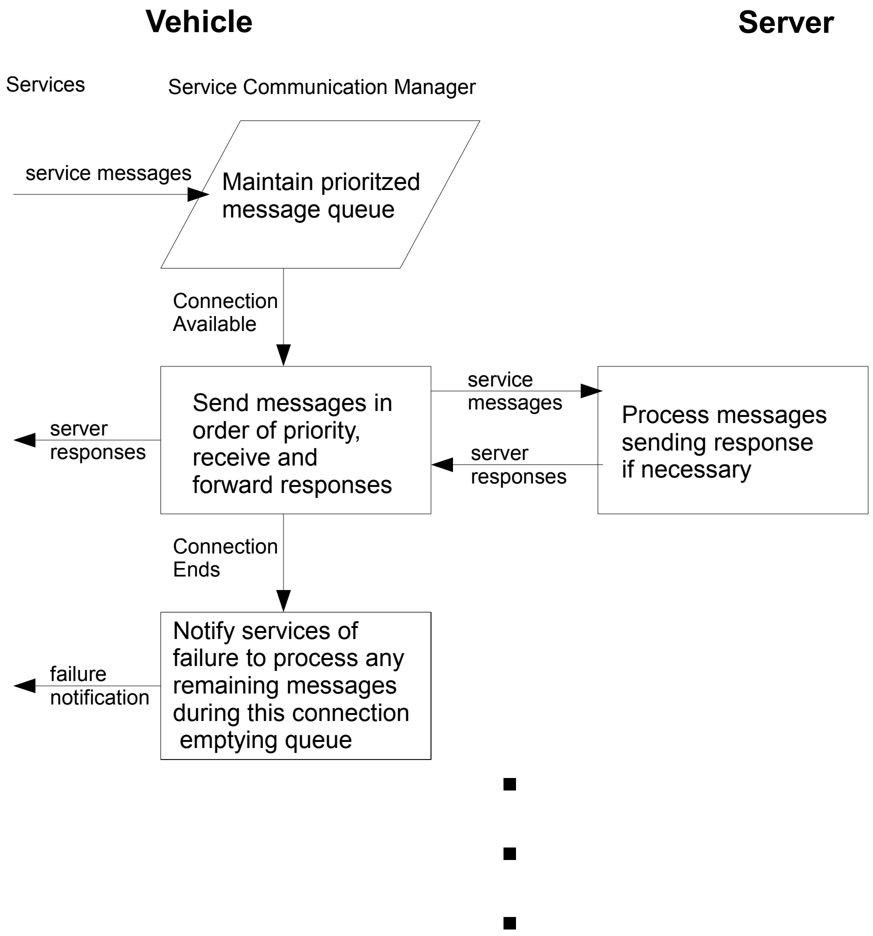


Figure 6.6: One iteration of the prioritised data exchange algorithm

offerings like Windows and Linux to more recent arrivals which are specifically designed for mobile devices such as Symbian, MeeGo and Android. At a block diagram level the service communications manager may reside in different parts of these potential operating systems and the practical implementation may vary in complexity/difficulty.

6.4.2 Communications Technology

There are a wide range of technologies which can potentially be used for vehicle communication such as Wi-Fi, Bluetooth, UWB, ZigBee, DSRC 915 MHz and IEEE 802.11p Wireless Access in Vehicular Environments (the flavour of Wi-Fi designed specifically to work in the vehicular environment). The implementation of the prioritized data exchange algorithm very much depends on the communications technology as the different technologies have varied pre-existing levels of support for prioritisation.

In practice for this thesis the IEEE 802.11p communications technology is considered. This technology was designed specifically to meet the challenges of the vehicular environment such as communicating while travelling at speed and the need for prioritisation of services. The in-built support for prioritisation in IEEE 802.11p works as follows. IEEE 802.11p has 75 megahertz of spectrum available in the 5.9 GHz frequency range and this is divided into seven channels. Provisions are made for the prioritisation of services; safety applications are prioritised over less critical services by 802.11p on-board units. Roadside units advertise to on-board units ten times a second what applications it supports, on which channels. On-board units listen on the control channel, authenticate the roadside units digital signature, execute safety applications first, then switch channels and execute non safety application. Safety applications use the channels designated for safety and non safety applications use the service channels as shown in figure 6.7 When all applications have been executed the on-board units returns to listening on the control channel.

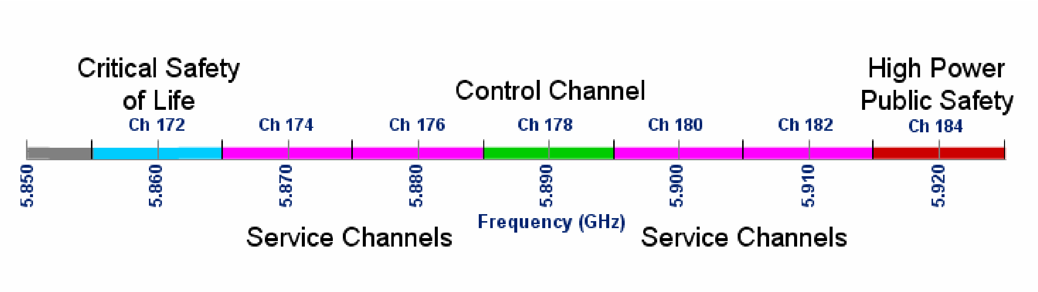


Figure 6.7: IEEE 802.11p channels: safety channels used to prioritise safety services [88]

When implementing the prioritised data exchange algorithm with IEEE 802.11p as the communications technology, the algorithm is not responsible for safety applications as this prioritisation is already catered for. The algorithm has responsibility for the remaining non safety applications such as traffic efficiency services and value added services such as infotainment. Looking at the algorithm description in figure 6.6 a connection is considered to be available when the IEEE 802.11p prioritization mechanism has finished prioritizing safety applications and switches from a safety channel to one of the service channels.

6.4.3 Assigning Priorities

Another important aspect of the implementation is the assigning of priorities to services. This is in essence a policy decision on which services are deemed to be the most important. When implementing the prioritised data exchange algorithm a decision must be taken on who gets to assign service priorities. In a system run by a third party individual users could be allowed to set there own service priorities or the third party may assign their own set of priorities to the services running. Alternatively the third party may set the priority of some services to be high e.g. safety services considered essential and leave the assignation of priority amongst the remaining services to the user.

In practice for this thesis the Smart City system proposed is designed to run services such as the traffic management service which work for the greater good of the city and all its inhabitants. So all service priorities are set by the maintainers of the city in keeping with this "by the city for the city" philosophy. In reality these maintainers are likely to be the city engineers and urban planners found in the offices of the city manager or some equivalent depending on the style of city governance.

6.5 Chapter Summary

In this chapter the algorithms associated with this thesis were presented. These were the best route selection algorithm used by the route management service described in chapter 4 and the prioritized data exchange algorithm used by the service communications manger outlined in chapter 5. High-level descriptions of the algorithms are given and algorithm implementation is also discussed.

Chapter 7

Experimental Tests

This chapter presents the experimental tests that were performed in relation to the proposed Best Route Selection Algorithm and the Prioritised Data Exchange Algorithm. Detailed descriptions of the test setups are given and in-depth analysis of the results that were obtained is performed. The presented results will prove the effectiveness of the proposed algorithms.

Due to the prohibitive expense of deploying a large scale vehicular network for test purposes it is necessary to have powerful simulation tools in order to test protocols, applications, algorithms etc., in the vehicular communications space. Two Simulation tools were used for the experimental tests in this thesis; SWANS++ [14, 33] and TraNS [128, 127]. Before the test results are presented, a brief introduction to some of the simulation tools available for vehicular communication is given in the next section.

7.1 Simulation Tools for Vehicular Communications

Simulation tools for vehicular communication must marry elements of transport science simulation and network simulation to accurately model the vehicular mobility and wireless communications involved.

Transport science classifies the models used to simulate road traffic into macroscopic, mesoscopic and microscopic models, based on performance functions and traffic flow representations as shown in table 7.1 [29]. Macroscopic models model traffic at a high level by approximating traffic as a fluid and making use of hydrodynamic flow theory. Individual vehicles cannot be explicitly traced and aggregate performance measures are obtained. Mesosopic models are concerned with the movement of groups of vehicles. Individual vehicles can be traced but performance measures are aggregated. Microscopic models work at the level of individual vehicles and provide precise position and performance measures for individual vehicles. *Vehicular network communications* are concerned with radio wave-based transmissions to and from individual vehicles, consequently only microscopic models give the level of precision necessary to accurately model vehicular communication.

		Performance Functions	
		Aggregate	*Disaggregate*
Flow Representation	*Continuous*	MACROSCOPIC	-
	Discrete	MESOSCOPIC	MICROSCOPIC

Table 7.1: Classification of road traffic models

An early approach to simulating communication in vehicular networks was to first generate a trace file of vehicle movements using a microscopic road traffic simulator. The trace file can then be used to drive the movement of mobile nodes in an existing network simulator such as ns-2. In order to mimic real world vehicle movements, microscopic road traffic simulators use some form of widely excepted micro-simulation model, e.g. the car following model [98], Intelligent-Driver/MOBIL Model (IDM/MOBIL) [159] or the Cellular Automaton (CA) model [120]. There are a number of simulation environments available that can generate trace files of realistic vehicle movements on real world roads; commercial examples include Daimler Chrysler's FARSI and VISSIM [131] from PTV AG, but there are also some freely available alternatives such as MOVE [93] and VanetMobiSim [66, 57]. It is also possible to generate a trace file by monitoring the movements of actual vehicles in the real world [102, 68]. A typical approach in this case is to equip a fleet of vehicles, which move around a city environment e.g. a taxi fleet, with GPS modules capable of recording GPS traces of their movement. The resulting data is then processed so it can be read by a network simulator.

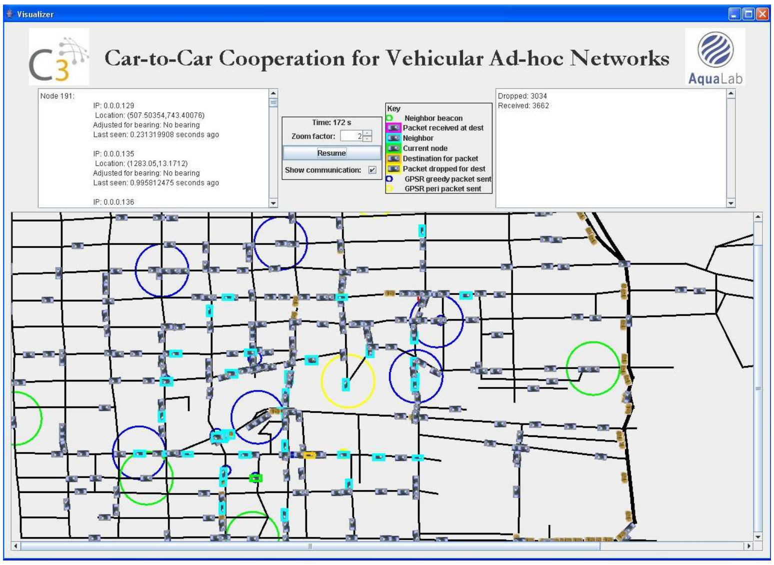


Figure 7.1: Visualisation of vehicular mobility and communication in SWANS++

However, in order to accurately simulate more complicated vehicular network scenarios such as the operation of traffic management type applications, it is necessary to employ a simulator capable of realistically emulating both, vehicular mobility and wireless communication, i.e. a simulator that integrates both traffic and network simulators as seen in figure 7.1. There are a number of open source simulators with this capability including -

- GrooveNet [106]: GrooveNet[1] is developed at the Carnegie Mellon University Department of Electrical and Computer Engineering in association with General Motors. GrooveNet enables the modelling of inter-vehicular communication within a real street map-based topography. It facilitates protocol design and also in-vehicle deployment. GrooveNet is a hybrid simulator which enables communication between simulated vehicles, real vehicles and between real and simulated vehicles. In the hybrid case, (i.e. communication between simulated vehicles and real vehicles on the road) the position, direction and messages of simulated vehicles are broadcast over the cellular interface from one or more infrastructure nodes. Real vehicles communicate with only those simulated vehicles that are within its transmission range.

[1]GrooveNet: http://www.ece.cmu.edu/ yonghooc/groovenet/

- NCTUns 4.0 [172]: NCTUns[2] (NCTU network simulator) is developed at Department of Computer Science, National Chiao Tung University, Taiwan. This simulator is a fully integrated simulation platform which combines the simulation capabilities provided by a network simulator and a traffic simulator.

- SWANS++ [14, 33]: SWANS++[3] (Scalable Wireless Ad-hoc Network Simulator) is developed at AquaLab, Department of Electrical Engineering and Computer Science, Northwestern University. This simulator supports IEEE802.11 based vehicular communication and realistic vehicular mobility on real world roads based on the car following model. SWANS++ includes support for some existing protocols suitable for vehicular communication, e.g. Greedy Perimeter Stateless Routing (GPSR) [94]. The current implementation SWANS++ alpha uses the STRAW traffic simulation module to support realistic vehicular mobility.

- TraNS [128, 127]: TraNS[4] (Traffic and Network Simulation Environment) is developed at the Laboratory for computer Communications and Applications (LCA), School of Computer and Communication Sciences, EPFL, Switzerland. In TraNS the information exchanged in vehicular communication protocols can influence the vehicle behaviour in the mobility model. For example, it is possible to model the fact that if a vehicle broadcasts information reporting an incident such as a multi-car collision, some of the neighbouring vehicles will slow down. The Current implementation TraNS v1.0 uses the SUMO traffic simulator and ns-2 network simulator. Support for realistic IEEE 802.11p is possible using ns-2.33.

- Veins [146]: Veins[5] (Vehicles in Network Simulation) is developed at the Department of Computer Sciences, University of Erlangen-Nuremburg, Germany. Veins is a vehicular communication simulation framework composed of an event-based network simulator and a road traffic microsimulation model. The two domain models are bi-directionally coupled and simulations performed on-line. This way, not only

[2]NCTUns 4.0: http://nsl10.csie.nctu.edu.tw/
[3]SWANS++: http://www.aqualab.cs.northwestern.edu/projects/swans++/
[4]TraNS: http://trans.epfl.ch
[5]Veins: http://www7.informatik.uni-erlangen.de/ sommer/veins/

can the influence of road traffic on network traffic be modelled, but the influence of vehicular communication on road traffic can also be modelled and complex interactions between both domains examined. The current implementation uses the SUMO traffic simulator and OMNeT++ network simulator.

7.2 Best Route Selection - Testing

The results from the analysis of the best route selection algorithm are presented in sections 7.2.3 through 7.2.8. The evaluation of the algorithm is divided into a number of stages. In these stages the following aspects are examined in order to determine the overall effectiveness of the algorithm:

- Comparison with alternative approaches
- The impact of the adaptive aspect of the algorithm
- The impact of the algorithm's k parameter
- The importance of the fitness function's cost components
- The effect of different road layouts on the algorithm
- The effect of reducing the penetration of the solution
- Investigation into global improvements versus individual improvements

The experiments are carried out using simulation based testing. The test setup is described in the next section.

7.2.1 Test Setup

In order to evaluate the best route selection algorithm simulations were performed with the Scalable Wireless Ad-hoc Network Simulator (SWANS) [14, 33]. This simulator supports realistic vehicular mobility modelling on real world roads. For testing real world road maps were used such as the subset of the road network of Boston, Ma, USA as highlighted in the

Figure 7.2: Sub network of Boston road network used marked with dotted line

figure 7.2 which was used for experiments in section 7.2.3 and section 7.2.4. The version used was SWANS++[6], this version contains the necessary vehicular mobility models and a visualisation tool. A typical SWANS simulation architecture is shown in figure 7.3. In this example it can be seen that in the wireless communication model TCP is used at the transport layer, IP version 4 at the network layer, IEEE 802.11b at the MAC layer and the Greedy Perimeter Stateless Routing (GPSR) protocol [94] is used for routing. The STRAW street mobility model [33] is used to model vehicular mobility. This mobility model uses a car following based micro simulation model that provides realistic vehicle behaviour including models for vehicle interactions with infrastructure elements such as traffic lights, stop signs, access ramps etc. The road network information is taken from real world road map sources such as the US Census Bureau's TIGER/Line files [7].

The individual test scenarios are outlined in detail for each experiment in the following sections. The traffic management solution, which uses the best route selection algorithm, will be referred to as the TraffCon solution in these sections for brevity. There is analysis of the test results for each experiment and finally the overall conclusions which can be derived from these results is summed up in the summary of results section. The simulation

[6] SWANS++ sourceforge project page: http://sourceforge.net/projects/straw/

[7] The US Census Bureau's TIGER/Line files:http://www.census.gov/geo/www/tiger/

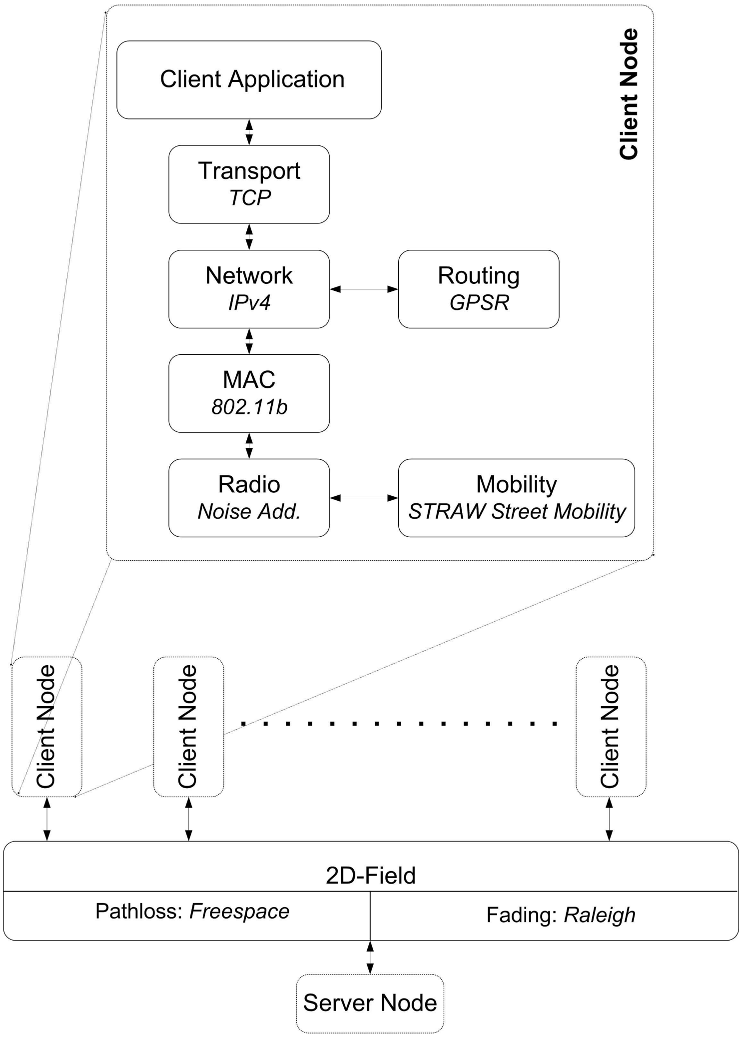


Figure 7.3: SWANS simulation architecture

settings of the SWANS++ simulator are discussed in detail in section 7.2.1.1 some important assumptions and parameters are first discussed hereafter.

As mentioned the SWANS simulator uses digital map data from real world roads. This data describes the road layout and contains information on speed limits, stop signs, traffic lights,access ramps etc. A vehicular mobility model is used to simulate real-life vehicle behaviour such as interaction between vehicles and at traffic lights. The simulator has the following provisions for repeatability, which allow the comparison of routing solutions.

Origin-Destination pairs are generated by the simulator. Vehicles travel from origin to destination obeying the mobility model in place, when a journey is completed a new origin destination pair is generated and a new vehicle starts a journey. The seed value used to generate the pairs is a configurable runtime parameter. Using the same seed value ensures the set of Origin-Destination pairs is repeatable. There is also a configurable runtime parameter for the maximum number of vehicles travelling in the map. The number of vehicles travelling in the system never exceeds the value of this parameter.When a routing solution is being tested vehicles will travel from origin to destination along the route that solution chooses. In order to test how the solution performs as traffic levels increase, a large number of runs are made incrementally increasing the number of vehicles present in the road network until it becomes saturated with traffic. For example the number of vehicles may be increased from 20 to 2000 in steps of 20. To reduce the influence of noise this set of runs is repeated a number of times using different seeds and results are averaged.

The timing settings of the simulations are also an important consideration. In practice for the following experiments a simulation time of two hours was used. This means that each run for a set number of vehicles examined a two hour period and that the configurable runtime parameter for simulation time was set to 7,200 seconds in all cases. To ensure that the simulation models did not start empty it was also necessary to warm-up the simulation model for each run. In practice it was found that a warm-up time of twenty minutes was sufficient and the configurable runtime parameter for warm-up time was set to seconds 1200 in all cases.

The TraffCon solution has a number of settable parameters these are; the value of k, the first k shortest paths by distance are compared using the fitness function, the values of w1, w2 and w3 the weights for the journey time, congestion and fuel consumption components of the fitness function and whether or not the solution adapts over the course of a journey may be switched on or off. In the following experiments different variants of TraffCon solution will be compared with other approaches and with each other. The description of individual experiments begins in the next section.

7.2.1.1 SWANS++ settings for simulation

When performing vehicular network simulations using the Scalable Wireless Ad-hoc Network Simulator[1] (version SWANS++) there are a variety of parameters which must be set to use a certain road network, follow a particular mobility model, mimic a particular technology etc. SWANS++ currently supports a large number configuration parameters that can be set at runtime. For the most part, runtime configurations are provided for elements which are most commonly changed and leave the rest as part of the compilation unit. Import settings are presented in detail hereafter.

Map Settings

In order to simulate vehicular mobility a digital road map is required. As well as describing road layout this map will contain information on speed limits, stop signs, traffic lights, access ramps etc. The parameters related to the road map used by the simulation and some sample values are shown in the following table.

Setting	Value	Note
segmentFile_	suffolk/segments.dat	Location of the segment file one of three files required to build the map
streetFile_	suffolk/names.dat	Location of the street file one of three files required to build the map
shapeFile_	suffolk/chains.dat	Location of the shape file one of three files required to build the map
minLat_	42.36134	The minimum latitude for the area of map to be used
minLong_	-71.05870	The minimum longitude for the area of map to be used
maxLat_	42.36808	The maximum latitude for the area of map to be used
maxLong_	-71.05024	The maximum longitude for the area of map to be used. A subset of the whole map may be used if the min/max values of latitude and longitude fall within the overall map area

Table 7.2: SWANS map settings for simulation

The digital map format used, which describes the map with three .dat files, comes from the Tiger Mapping and Routing Server[2] (TMRS). TMRS can convert data from TIGER, ESRI, GDT and Navteq map sources into its own file format. In practice for this thesis data from the US Census Bureaus TIGER/Line[3] files was used. This data is freely available on

[1] http://www.aqualab.cs.northwestern.edu/projects/swans++/
[2] http://www.sumitbirla.com/software/tmrs.php
[3] http://www.census.gov/geo/www/tiger/

the US Census Bureau website and maybe found as follows. The US Census Bureau map data is first indexed by feature, followed by state and then county, so to find road data for a particular City one must first select the Roads feature and then the state and the county where that city can be found. Take for example the city of Boston from which map data was taken for use in this thesis, to download data for the city of Boston one must first select the Roads feature, then the state of Massachusetts and finally Suffolk County. The other city from which map data was taken in this Thesis was Chicago which can be found in Cook County in the state of Illinois.

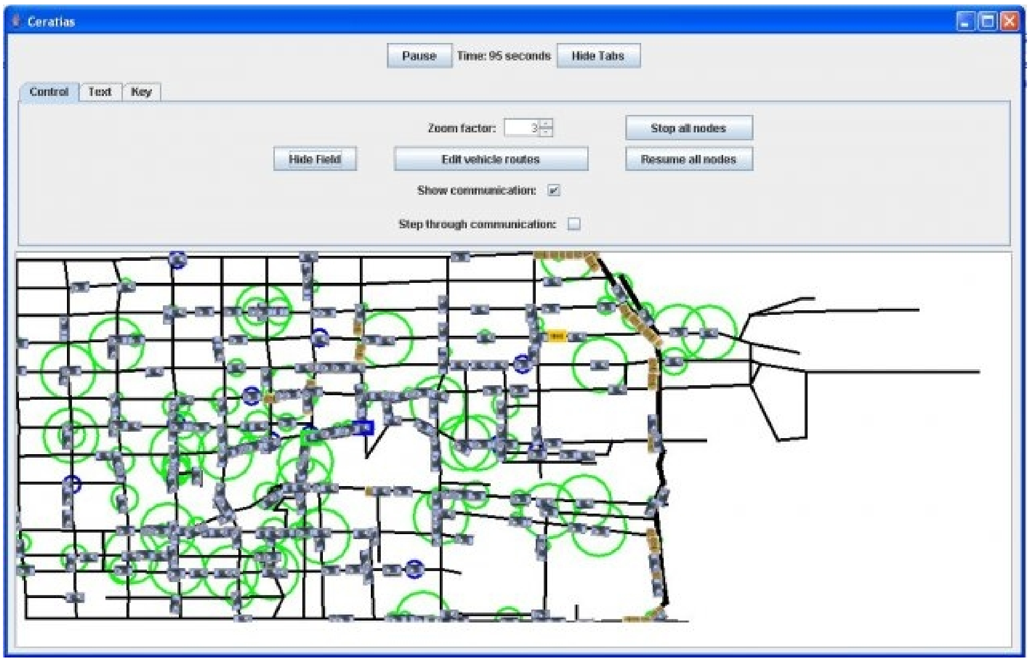


Figure 7.4: SWANS++ runtime visualisation tool

Simulation Settings

XML files are used to configure runtime parameters in SWANS, (a sample XML configuration file and further hints on modifying SWANs are given at the end of this thesis in Appendix B). Some important parameters are shown in the table 7.3.

[1] http://www.aqualab.cs.northwestern.edu/projects/STRAW/

Setting	Value	Note
duration_	900	Simulation time in seconds
resolutionTime_	60	Resolution period for simulation (seconds)
startTime_	60	Warm up period for simulation (seconds)
mobility_	6	The mobility model to be used by the simulator for example 6 represents the STRAW[1] vehicular mobility model while 2 represents a random way point mobility model
nodes_	100	Maximum number of vehicles in the road network at any one time
seed_	1234561	Origin-Destination pairs are generated by the simulator. Vehicles travel from origin to destination according to the mobility model in place, when a journey is completed a new origin destination pair is generated and a new vehicle starts a journey. The number of vehicles travelling in the system never exceeds the value of nodes_. Using the same seed value ensures the set of Origin-Destination pairs is repeatable
placement_	3	The placement model to be used for initial vehicle placement, for example 3 uses the STRAW street placement model while 1 is a random placement model
useVisualiser_	false	switches use of visualisation on or off, if set to true the simulation can be viewed in the runtime visualisation tool as seen in figure 7.4. Simulation runs more quickly with visualisation switched off

Table 7.3: SWANS simulation settings

7.2.2 Comparison with alternative approaches

Test Scenario

In order to evaluate the efficiency of the proposed TraffCon solution the following experiment was performed examining three different scenarios, which involve TraffCon and two competing approaches. In order to reduce the influence of noise in the results, the experiments were run three times using different seeds and the results were averaged.

Case (1): before each vehicle embarks on its journey it selects a shortest route using the A* shortest path algorithm [63]. The vehicle does not deviate from this route. The shortest path algorithm weights the road segments along a path according to the estimated time taken to traverse them. This is calculated by multiplying the length of the segment by the speed limit on that segment and adding a time penalty based on the intersection type found on the segment. This solution is referred to as A* based on distance hereafter.

Case (2): before each vehicle embarks on its journey it selects a shortest route using the A* shortest path algorithm. In this instance however the shortest path algorithm weights the road segments along a path using the actual time taken to traverse them as measured by vehicles travelling on the road network. These time values are gathered in the same manner as for the TraffCon solution. This solution is referred to as A* based on time hereafter.

Case (3):each vehicle drives to its own destination according to the route management solution with dynamic adaptation during the journey. The weights of the fitness function are set at w_1 = .7 , w_2 = .2 and w_3 = .1 and k was set to 20

The simulation time was set at two hours. In each simulation, the number of vehicles on the road was varied and average journey time, number of completed journeys, fuel economy was measured.The origin-destination pairs are kept constant i.e. the same vehicles are attempting to complete the same journeys in all cases. Border behaviour is not an issue as journeys occur within the simulation area. Whenever a journey is completed, another vehicle commences a journey, so the number of vehicles is maintained constant during the whole simulation duration.

Results and Analysis

The results for average journey time are compared in figure 7.5. In the first portion of the graph all solutions behave similarly as there are a very small number of vehicles present on the road network and traffic congestion is not an issue. As the number of vehicles increases average journey time begins to increase for all solutions. It can be seen in the graph that the TraffCon solution keeps journey time significantly lower than A* based on time as the number of vehicles increases. Both solutions keep journey time lower than A* based on distance. Eventually as the number of vehicles is increased the road network becomes saturated with vehicles and will become gridlocked. If vehicles cannot move they cannot complete their journeys and average journey time will approach infinity. This trend is seen in the graph but as the increase in vehicles sends the road network towards gridlock the TraffCon solution remains the best performing.

Given that the TraffCon solution exhibits lower average journey times for vehicles you would expect more journeys to be completed in a two hour window with that solution than any other and this is borne out in figure 7.6. We can see again in this graph the eventual trend toward gridlock as vehicle numbers are increased. At first completed journeys begins to rise steeply, then they begin to level off before they peak and begin to drop off, approaching zero

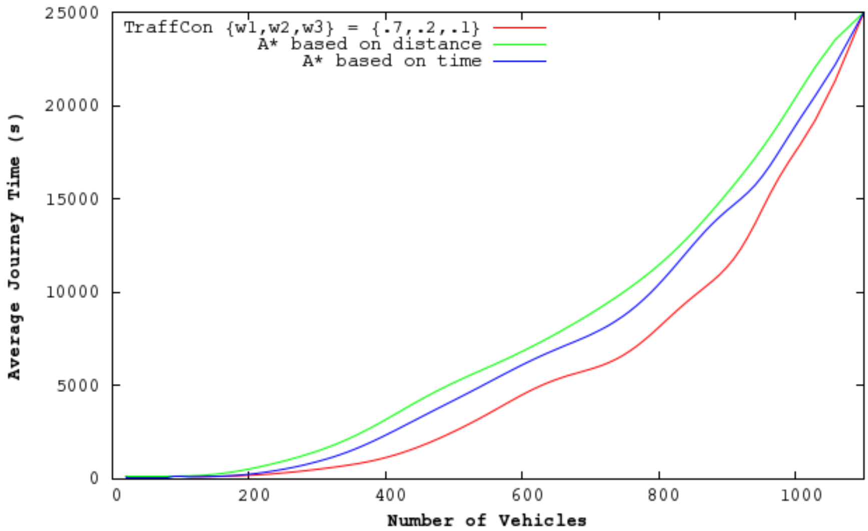


Figure 7.5: Average journey time against number of vehicles

too many vehicles travelling on the road network causes traffic to grind to a halt. We can see that as vehicle numbers rise the number of completed journeys for the TraffCon solution rises for longer, peaks higher and drops off more slowly than the A* solution based on time and the A* solution based on distance.

Looking more closely at the completed journeys graph we can see how much of an improvement the TraffCon solution achieves. The A* based on time solution peaks around the 200 vehicle mark, in figure 7.7 the number of completed journeys is shown at the 200, 300 and 400 vehicle mark. From this we chart we can see how the TraffCon solution compares with the A* based on time solution when the latter is at its peak and then beginning to drop off. At the 200 mark A* gives 10,124 completed journeys in two hours, the TraffCon solution totals 12,226 an extra 2,102 completed journeys or an increase of approximately 20.8%. At the 300 and 400 vehicle marks A* registers 5,234 and 2,499 completed journeys while TraffCon totals 10,439 and 6,389. That is an extra 5,205 completed journeys at the 300 vehicle mark and 3,890 at the 400 mark for increases of approximately 99.4% and 155.7%. The TraffCon solution significantly increases the peak value of completed journeys and significantly slows the decline in number of completed journeys in the drop

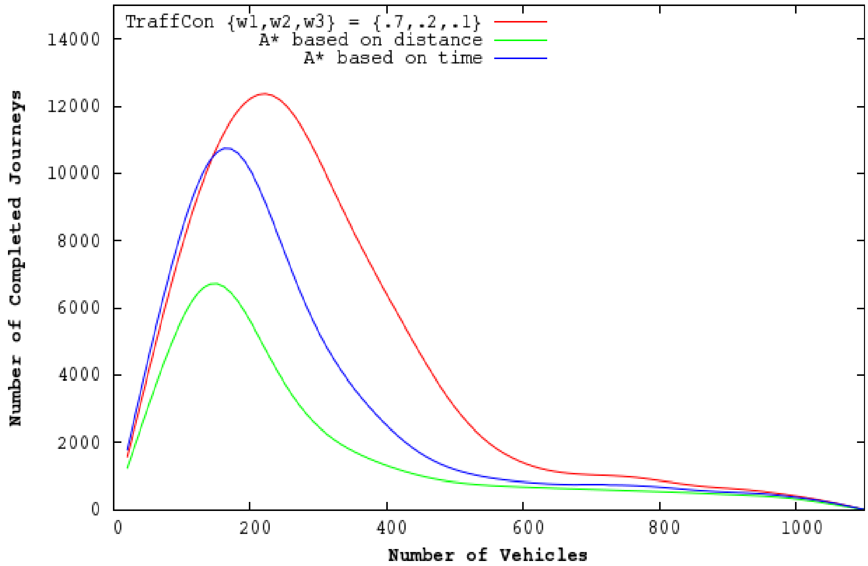
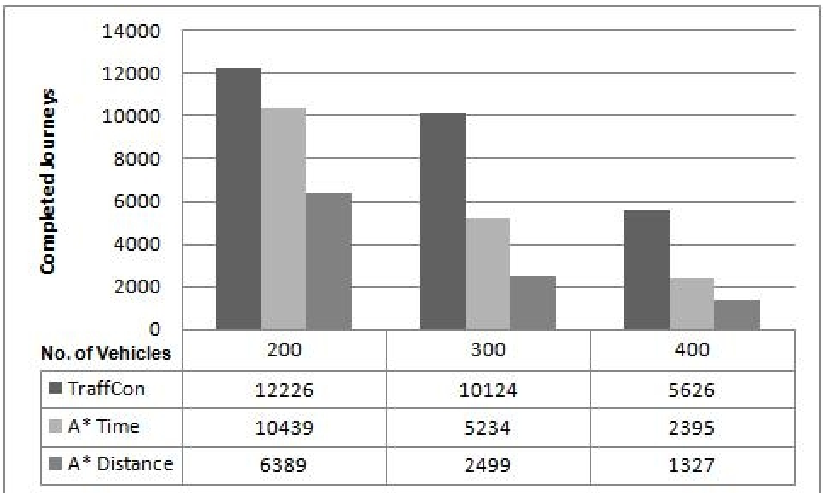


Figure 7.6: Number of journeys completed against number of vehicles

off prior to the road network becoming gridlocked. This means the TraffCon solution can push significantly more traffic through the road network than the A* solution or that it significantly increases the effective capacity of the road network when compared with the A* solution.



No. of Vehicles	200	300	400
TraffCon	12226	10124	5626
A* Time	10439	5234	2395
A* Distance	6389	2499	1327

Figure 7.7: Number of journeys completed for selected number of vehicles

The results for average fuel economy are compared in figure 7.8. In the first portion of the graph all solutions behave similarly as there are a very small number of vehicles present on the road network and traffic congestion is not an issue. As the number of vehicles increases average fuel economy begins to decline for all solutions. It can be seen in the graph that the TraffCon solution keeps fuel economy significantly higher than A* based on time as the number of vehicles increases. Both solutions keep fuel economy higher than A* based on distance. Eventually as the number of vehicles is increased the road network becomes saturated with vehicles and will become gridlocked. If vehicles cannot move they cannot complete their journeys and average fuel economy will approach zero. This trend is seen in the graph but as the increase in vehicles sends the road network towards gridlock the TraffCon solution remains the best performing.

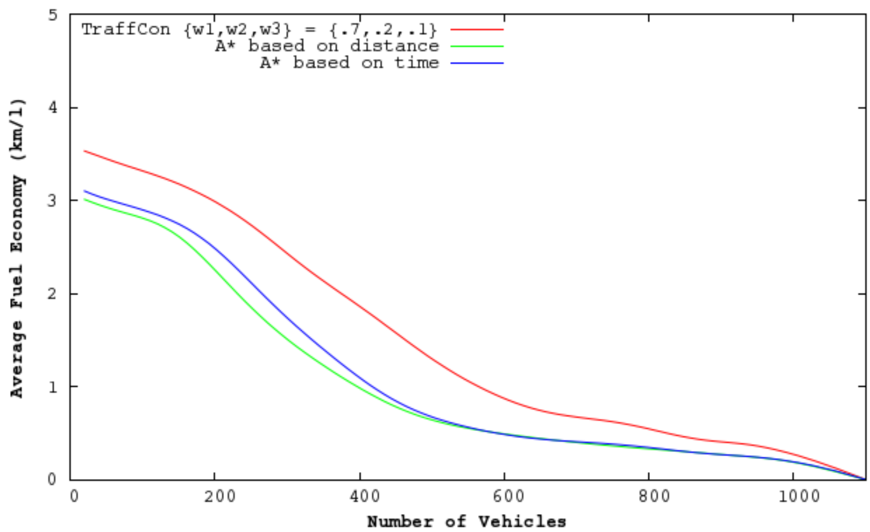


Figure 7.8: Average fuel economy against number of vehicles

In conclusion the TraffCon solution was shown to reduce journey times for drivers, increase the effective capacity of the road network and reduce fuel consumption compared with solutions that pick routes based on distance or time. The improvements were expected as the TraffCon solution uses a fitness function which considers journey time, congestion and fuel consumption components when picking the best route to take. How the various parameters of the TraffCon solution contribute to its performance will be examined in the following sections.

7.2.3 Investigation into algorithm adaptivity

In order to determine the importance of adaptation to the algorithm described in section 6.1 the performance of both non-adaptive and adaptive approaches are compared against an existing approach and against each other.

7.2.3.1 Non-adaptive algorithm

Test Scenario

In order to evaluate the efficiency of the proposed solution the following experiment was performed examining three different scenarios, which involved three competing approaches, including non-adaptive TraffCon.

Case (1): before each vehicle embarks on its journey it selects a shortest route using the A* shortest path algorithm [63]. The vehicle does not deviate from this route. The shortest path algorithm factors in the speed limit and a turn penalty based on intersection type for each road segment.

Case (2): each vehicle drives to its own destination according to the route management solution but without adaptation along the way. The weights of the fitness function presented in equation (6.5) are set at w_1 = 0.5, w_2 = 0.5 and w_3 = 0. Different values of k (5, 10, 15 and 20) are tested.

Case (3): results for a hypothetical ”ideal” solution are derived, where the solution performs well right up until the road network reaches vehicle saturation point i.e. length of available road divided by average vehicle length.

The simulation time was set at two hours. In each simulation, the number of vehicles travelling in the network was varied and average journey time, speed and fuel economy was measured. The origin-destination pairs are kept constant i.e. the same vehicles are attempting to complete the same journeys in all cases. Border behaviour is not an issue as journeys occur within the simulation area. Whenever a journey is completed, another

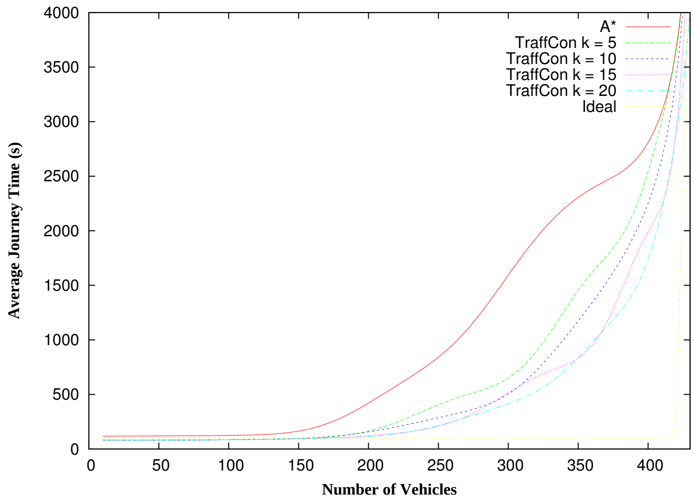


Figure 7.9: Graph of Avg. Journey Time(s) against Number of Vehicles

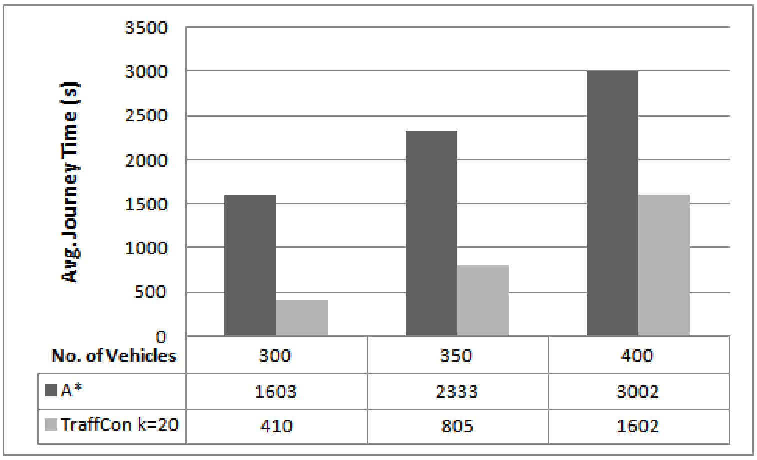


Figure 7.10: Avg. journey times(s) at selected vehicle numbers

vehicle commences a journey, so the number of vehicles is maintained constant during the whole simulation duration. In order to reduce the influence of noise in the results, the experiments were run three times using different seeds and the results were averaged.

Results and analysis

The results for average journey time are compiled in figure 7.9. As more and more vehicles are travelling on the road network, it becomes increasingly congested causing the average journey time to increase until the system becomes completely gridlocked. In the graph congestion is occurring when average journey time begins to rise noticeably and gridlock when

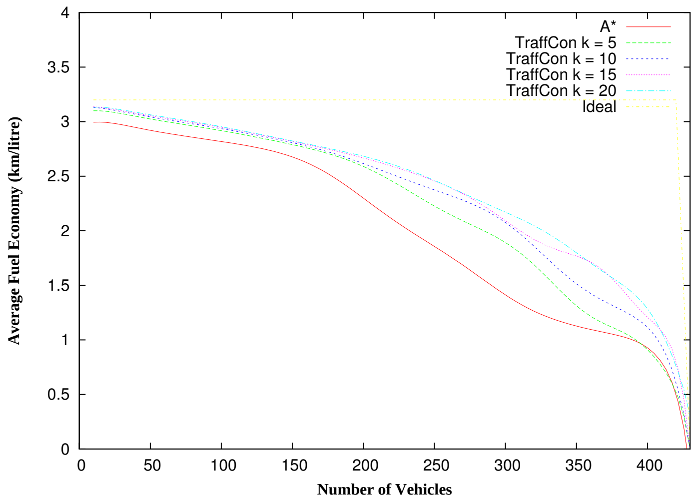


Figure 7.11: Graph of Avg. Fuel Economy (km/litre) against Number of Vehicles

the average journey time rises steeply. It can be clearly seen by observing those characteristics in the graph that the TraffCon solution reduces traffic congestion very significantly over the shortest path solution and even comes reasonably close to the unachievable ideal solution.

In figure 7.10 the performance of the shortest path solution and the best performing TraffCon solution (k=20) in the portion of the graph where congestion becomes evident are scrutinized. This is done by measuring average journey time when the number of vehicles is 300, 350 and 400 respectively. The bar chart clearly shows how much the shortest path solution A* is improved upon by TraffCon e.g. with 350 vehicles: average journey times are 2333 and 805 seconds respectively a 65% reduction when using TraffCon.

In figure 7.11 the results for average fuel economy are shown. Again it can clearly be seen that TraffCon improves over the shortest path solution. As more and more vehicles travel on the road network it becomes increasingly congested causing the average fuel economy to drop until the system becomes completely gridlocked and vehicles are burning fuel but not moving causing fuel economy in kilometres per litre to go to zero. The TraffCon solutions in figure 7.10 clearly give significantly higher fuel economies before they trend towards zero due to gridlock.

The results for average vehicle speed are compiled in figure 7.12. As more and more

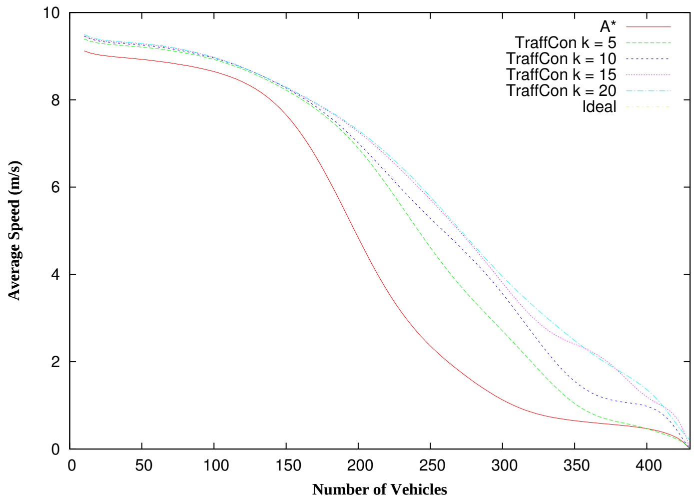


Figure 7.12: Graph of Avg. Speed(m/s) against Number of Vehicles

vehicles are travelling on the road network it becomes increasingly congested causing the average speed to drop until the system becomes completely gridlocked and vehicles are burning not moving. This gridlock point for a solution is reached when the average speed begins to descend steeply. The characteristics of this graph show that the reduction in journey time is being achieved in part by increasing the average speed over the shortest path solution as would be expected.

7.2.3.2 Adaptive Algorithm

Test Scenario

In order to evaluate the efficiency of the proposed solution the following experiment was performed examining four different scenarios, which involved four competing approaches, including TraffCon with static and dynamic route adaptation, respectively. In order to reduce the influence of noise in the results, the experiments were run three times using different seeds and the results were averaged.

Case (1): before each vehicle embarks on its journey it selects a shortest route using the A* shortest path algorithm [63]. The vehicle does not deviate from this route. The shortest path algorithm factors in the speed limit and a turn penalty based on intersection type for each road segment.

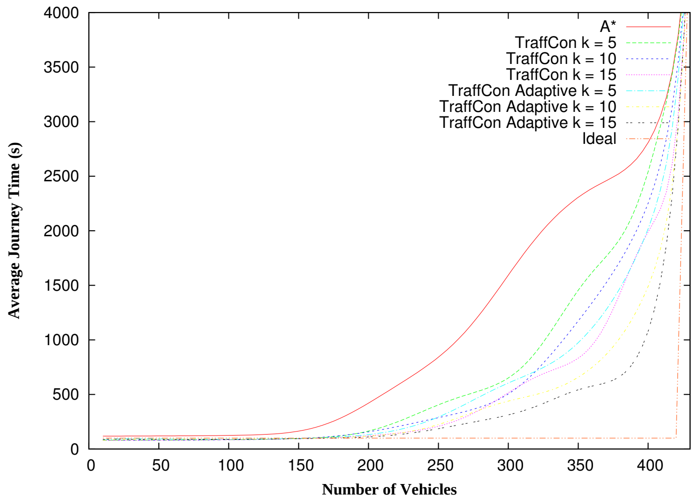


Figure 7.13: Average Journey Time(s) against Number of Vehicles in the Network

Case (2): each vehicle drives to its own destination according to the route management solution but without adaptation during the journey. The weights of the fitness function described in equation (6.5) are set at w_1 = 0.5, w_2 = 0.5 and w_3 = 0. Different values of k (5, 10 and 15) are tested.

Case (3):each vehicle drives to its own destination according to the route management solution with dynamic adaptation during the journey. The weights of the fitness function are set at w_1 = 0.5, w_2 = 0.5 and w_3 = 0 and different values of k (5, 10 and 15) are considered.

Case (4): results for a hypothetical "ideal" solution are derived, where the solution performs well right up until the road network reaches vehicle saturation point i.e. length of available road divided by average vehicle length.

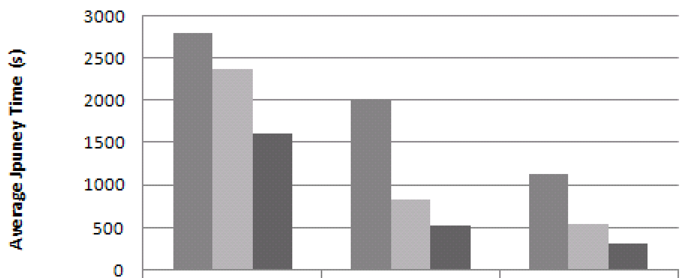
The simulation time was set at two hours. In each simulation, the number of vehicles on the road was varied and average journey time and fuel economy was measured.The origin-destination pairs are kept constant i.e. the same vehicles are attempting to complete the same journeys in all cases. Border behaviour is not an issue as journeys occur within the simulation area. Whenever a journey is completed, another vehicle commences a journey, so the number of vehicles is maintained constant during the whole simulation duration.

Results and analysis

The results for average journey times (expressed in seconds) are compiled in figure 7.13. As the number of vehicles travelling on the road network is increased for each solution, eventually average journey time begins to increase noticeably as the system becomes congested. At some point average journey time begins to climb steeply and the system is effectively gridlocked. The best performing non-adaptive TraffCon solution shown (k = 15) performs 44% better than the shortest path method as their gridlock points are approximately 250 and 360 vehicles respectively. The best performing adaptive TraffCon solution shown (k = 15) performs a further 9.7% better as it does not become gridlocked till there are 395 vehicles in the system. This is only 6.3% off the unachievable ”ideal” solution, which becomes gridlocked at 420 vehicles.

In figure 7.14 the performance of the shortest path solution and the best performing adaptive and non-adaptive TraffCon solutions (k=15) are examined in the portion of the graph where congestion is apparent by inspecting average journey time values when the number of vehicles in the system is 400, 350 and 300, respectively. Both the table and bar chart clearly show how the shortest path solution A* is improved upon by the non-adaptive TraffCon, which is in turn further improved upon by the adaptive TraffCon e.g. with 350 vehicles average journey times are 2370, 835 and 555 seconds respectively.

The results for average fuel economy (km/litre) are compiled in figure 7.15. As more



No. of Vehicles	400	350	300
A*	2810	2020	1130
TraffCon k=15	2370	835	555
TraffCon Adaptive k=15	1610	540	315

Figure 7.14: Average journey times(s) for selected vehicle numbers

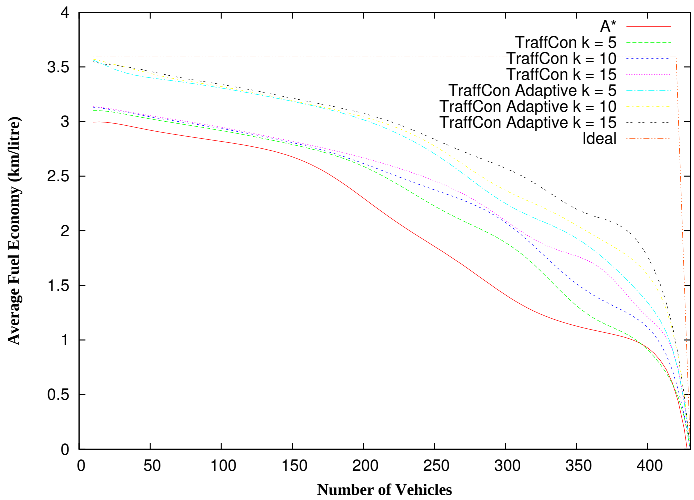


Figure 7.15: Average Fuel Economy (km/litre) against Number of Vehicles

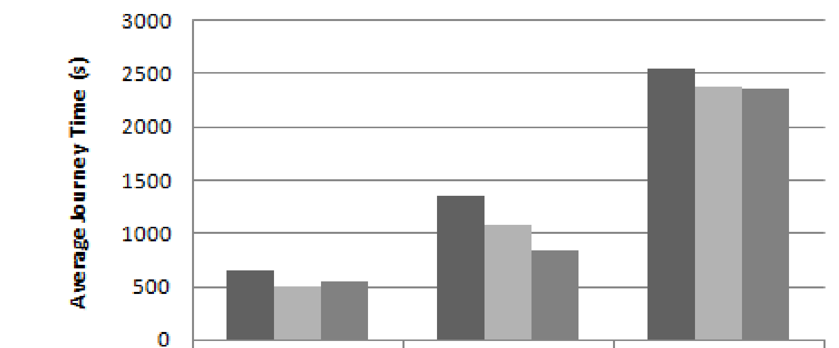
and more vehicles are travelling on the road network it becomes increasingly congested causing the average fuel economy to decrease until the system becomes completely gridlocked and vehicles are burning fuel but going nowhere. This gridlock point for a solution is reached when the average fuel economy begins to descend steeply. The best performing non-adaptive and adaptive TraffCon solution shown are when k = 15. Non-adaptive TraffCon performs 32.7% better than the shortest path method as their gridlock points are approximately 245 and 325 vehicles respectively. Adaptive TraffCon performs a further 20% better as its gridlock point is approximately 390 vehicles. This is 7.7% off the unachievable ”ideal” solution which has a gridlock point of 420 vehicles.

It can be seen in figures 7.13 and 7.15 that the significant improvements made by TraffCon over the shortest path solution occur over a wider range for fuel economy than journey time. This is because fuel consumption depends not only on journey time but also on velocity and acceleration associated with the route. The tested implementation reduces the ”stop-startiness” which occurs when vehicles impede one another and reduces fuel consumption by evenly distributing the density of vehicles on individual road segments.

In conclusion variants of the TraffCon solution that do not adapt perform well but having adaptivity switched on improves performance. The performance of the adaptive TraffCon solution for a given value of k is better than the corresponding non-adaptive solution in all cases. Where possible implementations of the TraffCon solution should adapt but if the level of adaptivity is forced to decline then the solution will still perform adequately. The availability of communication hotspots is the main limit on the ability to adapt, if there is a hotspot at every intersection then the solution can run with full adaptivity a density of hotspots less than this for example a density of a hotspot at every second intersection would support less than full adaptivity.

7.2.4 Investigation into the impact of the k parameter

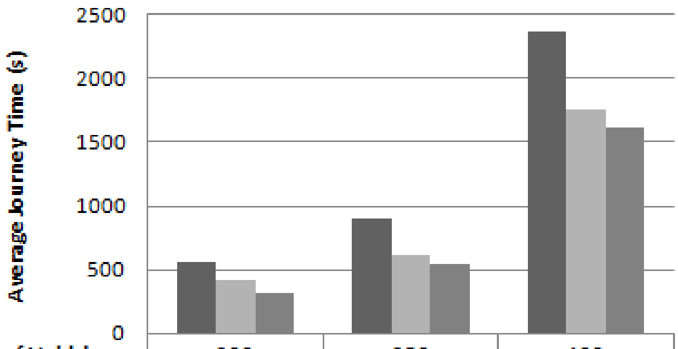
The impact of the k parameter can be investigated using the results in the previous section. It is obvious from figures 7.13 and 7.15 that the performance of the TraffCon solution improves as the value of k is increased in both the non-adaptive and adaptive cases as expected.



No. of Vehicles	300	350	400
TraffCon k=5	650	1350	2560
TraffCon k=10	500	1075	2390
TraffCon k=15	555	835	2370

Figure 7.16: Average journey times(s) at selected vehicle numbers

This improvement in performance with increasing values of k is again highlighted in figures 7.16 and 7.17. In figure 7.16 the performance of non-adaptive TraffCon solutions with values of k set to 5,10 and 15 are examined in the portion of the graph where congestion is apparent by inspecting average journey time values when the number of vehicles in the system is 300, 350 and 400, respectively. Both the table and bar chart clearly show how the performance improves as the value of k increases e.g. with 350 vehicles the average



No. of Vehicles	300	350	400
Traffcon Adaptive k=5	560	900	2370
TraffCon Apdaptive k=10	420	610	1760
TraffCon Adaptive k=15	315	540	1610

Figure 7.17: Average journey times(s) at selected vehicle numbers

journey times when k is equal to 5, 10 and 15 are 1350, 1075 and 835 seconds respectively. As expected as k increases average journey times decrease. In figure 7.17 adaptive TraffCon solutions with values of k set to 5,10 and 15 are similarly examined. Again both the table and bar chart show how that performance improves as the value of k increases e.g. with 400 vehicles the average journey times when k is equal to 5, 10 and 15 are 2370, 1760 and 1610 seconds respectively.

Another observation can be made about the k parameter from figures 7.9, 7.11 and 7.12 other than the TraffCon solution improving as k increases as expected. That is that the improvement brought about by increasing k converges quickly as there is an upper bound enforced by the road network in use. This can be seen in figures 7.9, 7.11 and 7.12 as there is a large improvement when increasing k from 5 to 10 but progressively smaller improvements when going from 10 to 15 and then 15 to 20. Consequently in practice when implementing the route selection algorithm a relatively small value of k will suffice for good performance e.g. a k value of approximately 30.

7.2.5 Investigation into the effect of road layout on the algorithm performance

Test Scenario

As mentioned, Simulations are performed with the Scalable Wireless Ad-hoc Network Simulator (SWANS) [33]. This simulator supports realistic vehicular mobility modelling on real world roads. In earlier experiments the road network highlighted in figure 7.2 was used. For this experiment two road networks both approximately twice that size are used - a sub network of the road network of Boston, MA as highlighted in figure 7.18 and of Chicago, IL as highlighted in figure 7.19. Two distinct road networks were used to examine the impact of the road layout on performance - the Boston map is a non regular road network whereas the Chicago map is a regular "Manhattan grid" as seen in figure 7.20.

Figure 7.18: Sub-network of Boston road network used (highlighted red)

In order to evaluate the effect of different road layouts on the algorithm the following experiment was performed examining three different scenarios, which involved three competing approaches. In order to reduce the influence of noise in the results, the experiments were run three times using different seeds and the results were averaged. The experiment was performed twice once using the Boston street map and a second time using the Chicago street map.

Case (1): before each vehicle embarks on its journey it selects a shortest route using the A* shortest path algorithm [63]. The vehicle does not deviate from this route. The shortest

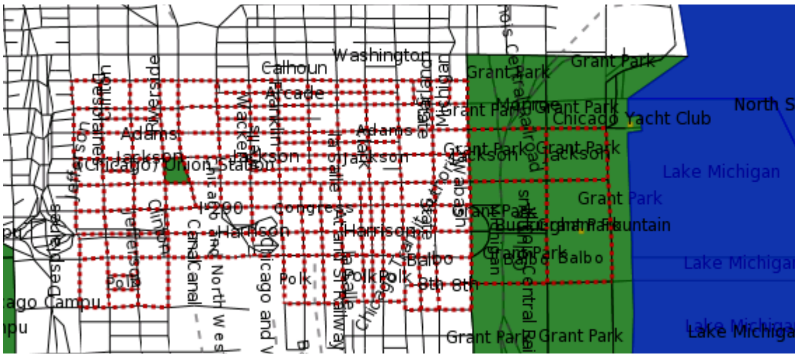


Figure 7.19: Sub-network of Chicago road network used (highlighted red)

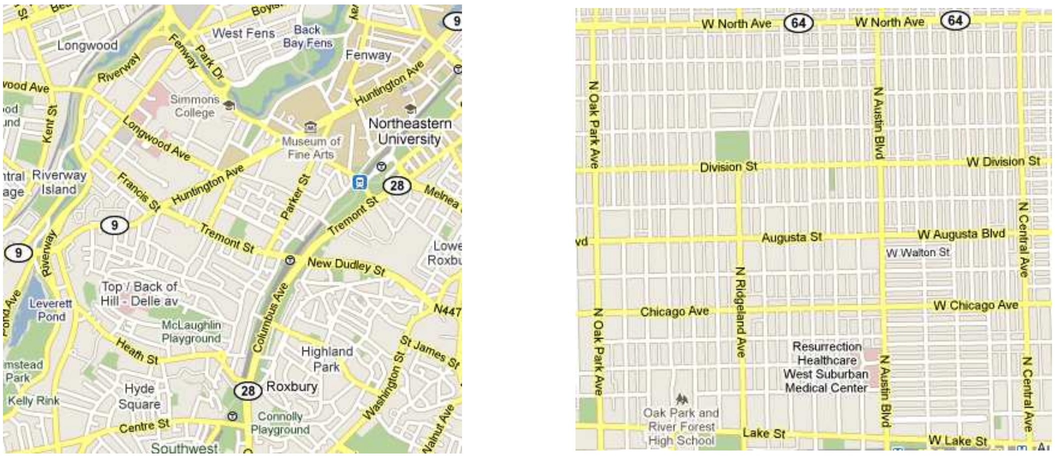


Figure 7.20: Boston map irregular network (left) & Chicago map regular grid (right)

path algorithm factors in the speed limit and a turn penalty based on intersection type for each road segment.

Case (2):each vehicle drives to its own destination according to the route management solution with dynamic adaptation during the journey. The k parameter was set at k = 15 and different values for the weights w_1, w_2 and $w_3$are considered.

Case (3): results for a hypothetical "ideal" solution are derived where suitable for a given parameter i.e. the solution is seen to perform well right up until the road network reaches vehicle saturation point i.e. length of available road divided by average vehicle length.

A representative test set of 43 distinct weight settings was used and the complete set

SET A

w1	w2	w3
1	0	0
0	1	0
0	0	1

SET B

w1	w2	w3
.8	.2	0
.6	.4	0
.5	.5	0
.4	.6	0
.2	.8	0

SET C

w1	w2	w3
.8	0	.2
.6	0	.4
.5	0	.5
.4	0	.6
.2	0	.8

SET D

w1	w2	w3
0	.8	.2
0	.6	.4
0	.5	.5
0	.4	.6
0	.2	.8

SET E

w1	w2	w3
.8	.1	.1
.6	.2	.2
.4	.3	.3
.3·	.3·	.3·
.2	.4	.4
0	.5	.5

SET F

w1	w2	w3
.1	.1	.8
.2	.2	.6
.3	.3	.4
.3·	.3·	.3·
.4	.4	.2
.5	.5	0

SET G

w1	w2	w3
.1	.8	.1
.2	.6	.2
.3	.4	.3
.3·	.3·	.3·
.4	.2	.4
.5	0	.5

SET H

w1	w2	w3
.7	.2	.1
.5	.3	.2
.7	.1	.2
.5	.2	.3

SET I

w1	w2	w3
.2	.1	.7
.3	.2	.5
.1	.2	.7
.2	.3	.5

SET J

w1	w2	w3
.2	.7	.1
.3	.5	.2
.1	.7	.2
.2	.5	.3

Figure 7.21: Set of weights examined during testing

can be seen in figure 7.21. The test set as shown in figure 7.21 is divided into a number of subsets A through J. Set A examines what effect each of the three weights has in isolation, sets B through D have one weight set to zero while the other two are varied ,in sets E through G the weights are varied such that two of the weights remain equal and in sets H through J one weight is made dominant while the relative strength of the other two is varied.

The simulation time was set at two hours. In each simulation, the number of vehicles on the road was varied and total completed journeys, and average, standard deviation in and best and worst individual speed and fuel economies were measured.

Total completed journeys is used to differentiate between traffic management solutions. The graph of completed journeys against number of vehicles should be similar to that of a Poisson distribution as seen in figure 7.22. The Poisson distribution has been widely used in transport science to model traffic arrival patterns [19, 105, 108] and the graph of total completed journeys against number of vehicles will represent a traffic arrival pattern. Three

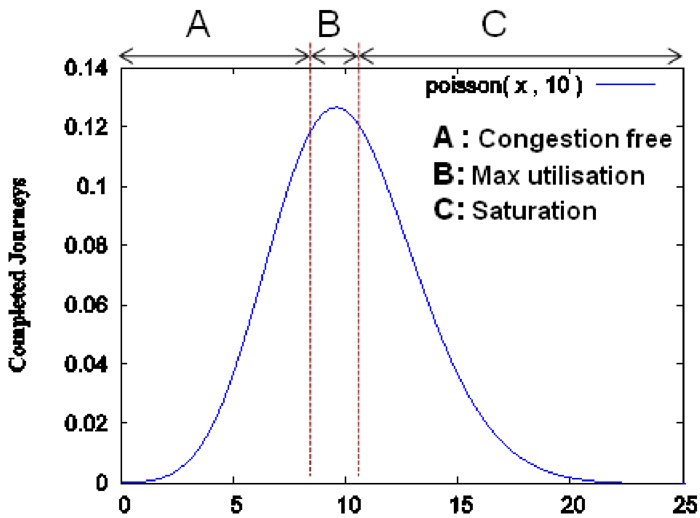


Figure 7.22: The Poisson distribution type curve expected for completed journeys against number of vehicles

main phases should be evident in each solution as represented by the 3 sections marked on the graph in figure 7.22. In section A of the graph there are not enough vehicles on the road for traffic congestion to be an issue and the number of completed journeys rises as the number of vehicles increases. In section B the graph flattens off around a peak value at this point the road network is at maximum utilisation. Finally in section C the road network becomes congested and the number of journeys completed decreases as the number of vehicles increases. Eventually the saturation point of the road network with vehicles is reached and the graph tails off to zero. The better the solution, the higher the peak the wider the flat portion around it and the longer the tail.

Results and Analysis

The experiment described in the previous section produced a very large set of results for the 43 TraffCon solutions using distinct weight settings all of which could not possibly be displayed here. However a systematic process of elimination was used to isolate a smaller set of the best performing solutions. In this section we will first describe the process of elimination and then analyse the set of best performing solutions.

The first measurement examined in other to reduce the solution set was the total number of journeys completed. On initial examination of the results for the Boston Map it was clear that the outliers were peaking around 12,000. The first step was to remove solutions below 75% of this observed upper bound i.e. those with a peak below 9,000. This removed the seven solutions in which the journey time cost was given a weight of zero. The next step was to remove all solutions that peaked below 12,000. After this 14 solutions remained and the 6 solutions with the highest peaks from that group are shown in figure 7.23.

From first observations of the results for the Chicago Map it was clear that the outliers were peaking about 20,000 completed journeys. As for the Boston map the first step was to remove solutions below 75% of the observed upper limit i.e. those with a peak below 15,000. This removed the seven solutions in which the journey time cost was given a weight of zero. From this we can see that the worst performing solutions for both maps are the same i.e. those which do not consider journey time cost. The next step was to remove all solutions that peaked below 20,000. After this 15 solutions remained, the set of 14 solutions remaining after this stage when examining the Boston map are a subset of the 15. From this we can see that the behaviour of the TraffCon solution is broadly similar irrespective of the road network i.e. whether the road network is irregular as in the Boston map or a "Manhattan grid" as found in the Chicago map. The 6 solutions with the highest peaks from the group of 15 are shown in figure 7.24.

For both maps the top two solutions are shown to be those with weights set to w_1 = 0.7, w_2 = 0.2,w_3 = 0.1 and w_1 = 0.8, w_2 = 0.1, w_3 = 0.1. This result further underlines that the behaviour of the best route selection algorithm is approximately the same irrespective of the road network i.e. whether the road network is irregular as in the Boston map or a "Manhattan grid" as found in the Chicago map. As seen in earlier experiments such as those in section 7.2.3 TraffCon solutions vastly improve on the A* shortest path solution. This can also be seen in figures 7.23 and 7.24. For example in figure 7.23 the shortest path solution peaks at 9615 completed journeys and the best TraffCon solution peaks at 16,152 an increase of 68%. In figure 7.24 the shortest path solution peaks at 9950 completed journeys and the best TraffCon solution peaks at 22,310 an increase of 124%. It is clear

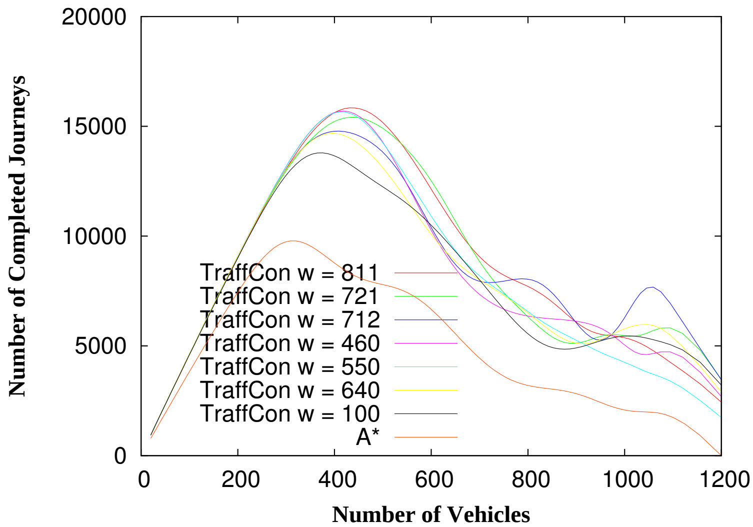


Figure 7.23: Completed journeys against number of vehicles for Boston map

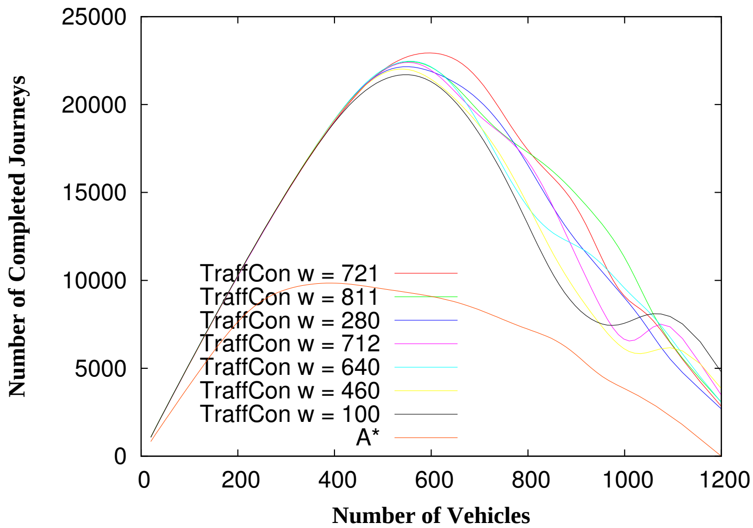


Figure 7.24: Completed journeys against number of vehicles for Chicago map

from this result that the TraffCon solution achieves performance improvements whether the road network is irregular as in the Boston map or a ”Manhattan grid” as found in the Chicago map.

7.2.6 Investigation into the fitness function’s cost components importance

The three cost components of the algorithm’s fitness function are Journey Time Cost, Used Capacity Cost and Fuel Consumption and Emissions Cost. The impact of the fitness function's cost components can be investigated using the results from the previous section. The process of eliminating the more poorly performing solutions is described in the previous section. In this process the first solutions to be eliminated were those that did not consider time i.e. those in which $w_1 = 0$. This shows that journey time component plays a very important role in producing a good solution. In this section we will demonstrate that the other two cost components also play an important role.

The results for number of completed journey for Boston and Chicago street maps respectively are shown in figures 7.23 and 7.24. They display the top 6 solutions from the set of 43 different weight settings examined in testing (outlined in figure 7.21) as well as the solution with weights set to $w_1 = 1$, $w_2 = 0$ and $w_3 = 0$ i.e. the solution where only time is considered. This was done so that solutions that use all 3 weighted cost components can be compared with a simpler approach where only time is considered.

For both maps the top two solutions are shown to be those with weights set to $w_1 = 0.7$, $w_2 = 0.2$,$w_3 = 0.1$ and $w_1 = 0.8$, $w_2 = 0.1$, $w_3 = 0.1$. This result shows that the best performing variants of the TraffCon solutions consider Journey Time Cost, Used Capacity Cost and Fuel Consumption and Emissions Cost. Averaging these two results to suggest the best weight settings predicts weight settings of 75%,15% and 10% for Journey Time Cost, Used Capacity Cost and Fuel Consumption and Emissions Cost to perform best. Out of the top six performing variants of the TraffCon solution all contain a time component and for four time is the largest component, all six contain a capacity component and three contain a fuel component. This like the 75%,15% and 10% weight ratios calculated above suggest that all components make a contribution but the order of importance is Journey Time Cost, Used Capacity Cost and Fuel Consumption and Emissions Cost.

In order to further understand the contribution of each of these components a series of new experiments were performed examining what happens when you minimise each component in isolation i.e. use weight settings of $w_1 = 1$, $w_2 = 0$,$w_3 = 0$ and $w_1 = 0$, $w_2 = 1$,$w_3 = 0$ and $w_1 = 0$, $w_2 = 0$,$w_3 = 1$.

Test Scenario

In order to evaluate the contribution of the individual components of the fitness function; Journey Time Cost, Used Capacity Cost and Fuel Consumption and Emissions Cost the following experiment was performed examining three variants of the TraffCon solution. In order to reduce the influence of noise in the results, the experiments were run three times using different seeds and the results were averaged.

Case (1): each vehicle drives to its own destination according to the route management solution with dynamic adaptation during the journey. The weights of the fitness function are set at w_1 = 1 , w_2 = 1 and w_3 = 0 and k was set to 20

Case (2): each vehicle drives to its own destination according to the route management solution with dynamic adaptation during the journey. The weights of the fitness function are set at w_1 = 0 , w_2 = 1 and w_3 = 0 and k was set to 20

Case (3):each vehicle drives to its own destination according to the route management solution with dynamic adaptation during the journey. The weights of the fitness function are set at w_1 = 0 , w_2 = 0 and w_3 = 1 and k was set to 20

The simulation time was set at two hours. In each simulation, the number of vehicles on the road was varied and average journey time and fuel economy was measured.The origin-destination pairs are kept constant i.e. the same vehicles are attempting to complete the same journeys in all cases. Border behaviour is not an issue as journeys occur within the simulation area. Whenever a journey is completed, another vehicle commences a journey, so the number of vehicles is maintained constant during the whole simulation duration.

Results and Analysis

The results for average journey time are compared in figure 7.25. In the first portion of the graph all solutions behave similarly as there are a very small number of vehicles present on the road network and traffic congestion is not an issue. As the number of vehicles increases average journey time begins to increase for all solutions. It can be seen in the graph that the variant of the TraffCon solution where w_1 = 1 out performs the other two substantially. This is hardly surprising as one would expect the variant that minimises journey time to produce

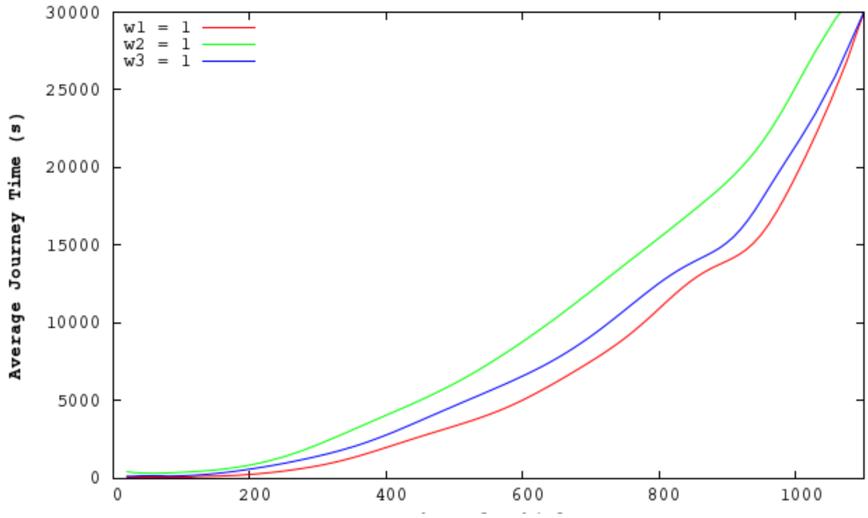


Figure 7.25: Average journey time against number of vehicles

lower journey times than the versions that minimise used capacity and fuel consumption.

The fuel consumption component considers time and measures the accelerations and decelerations of vehicles, which impact fuel consumption. Situations that cause a vehicle to brake and then accelerate again negatively impact fuel consumption, the kinetic energy gained by burning fuel is converted to heat by friction when braking and lost, it is regained by burning more fuel to accelerate back up to speed. The fuel consumption component is supposed to help avoid these situations. An existing fuel economy solution that attempts to minimise the effect of braking on fuel consumption is Kinetic Energy Recovery Systems (KERS), pioneered in motor sport this technique improve fuel economy by converting the kinetic energy into electric energy and storing it to be used later to power the vehicle. The variant with $w_3 = 1$ gives lower average journey times than that with $w_2 = 1$ as the fuel consumption component considers time but the used capacity component does not. The capacity component tries to minimises the used capacity of each road segment on its own this spreads vehicles out across the road network as much as possible but is not good for minimising journey time.

The results for average fuel economy in figure 7.26 tell us more about the used capacity and fuel consumption components. In this graph as expected you see that fuel economy

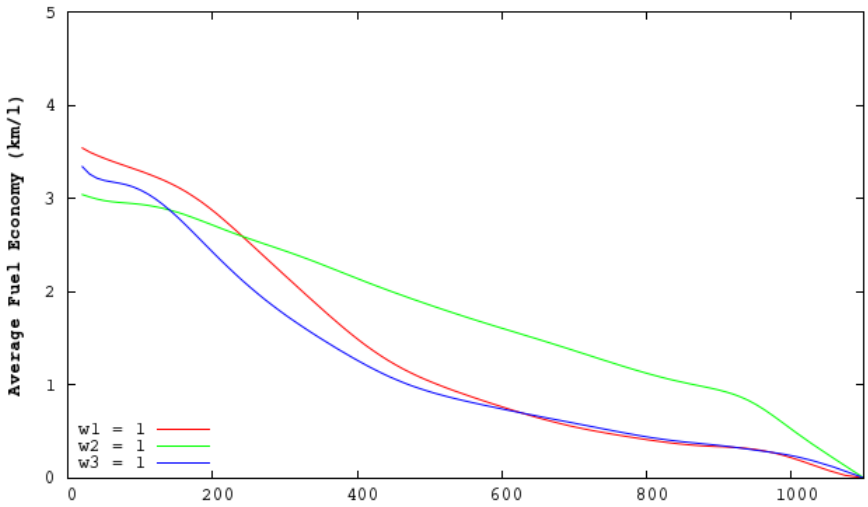


Figure 7.26: Average fuel economy against number of vehicles

starts high and heads towards zero as the number of vehicles increase and the road network becomes more congested. Surprisingly when with $w_3 = 1$ the solution does not perform better than the others initially and only provides marginally better fuel economy than the case where $w_1 = 1$ as the road network becomes congested. The variant of TraffCon where $w_2 = 1$ significantly outperforms the other two keeping fuel economy much higher as the network becomes increasingly congested from the 300 vehicle mark on.

In the urban environment two major contributors to fuel consumption are idling in traffic at intersections, and braking and accelerating due to situations encountered on the road network. The time component will help eliminate these to some extent as road segments with long waits at an intersection may take longer to cross. The fuel consumption component should detect idling and also instances of stop/start or brake/accelerate type behaviour and reduce fuel consumption by avoiding it. However many of the the situations which cause brake/accelerate behaviour in real life such as pedestrians crossing the road, cyclists, parked cars, speed bumps etc. are not modelled in the simulator consequently they are not as much of this behaviour for the fuel consumption parameter to minimise. The single biggest cause of brake/accelerate behaviour for a vehicle in the model is interactions with other vehicles for example yielding for other vehicles at intersections, slowing to follow a vehicle going more slowly, accelerating to overtake a slow moving vehicle etc. As the number of vehicles increases the number of interactions between vehicles should increase so you would expect the variant of TraffCon where w_3 = 1 to begin to outperform that where w_1 = 1 if the number of vehicles increases enough and this can be seen to be true in the graph from the 600 vehicle mark onwards.

The parameter that should best minimise the interaction between vehicles and hence the idling and brake/accelerate behaviour that impacts fuel economy is the used capacity cost. Minimising this parameter should result in a more even spread of vehicles across the road network making it less likely that vehicles interfere with each others progress. This parameter does work as expected because the variant of TraffCon where w_2 = 1 significantly outperforms the other two keeping fuel economy much higher as the network becomes increasingly congested from the 300 vehicle mark on.

In conclusion the three components of the fitness function; Journey Time Cost, Used Capacity Cost and Fuel Consumption and Emissions Cost behave as expected. Journey time is the best at minimising journey time, used capacity cost spreads vehicles more evenly across the road network and Fuel Consumption and Emissions Cost provides some benefit for improving fuel economy. The Fuel Consumption and Emissions Cost is not however as effective as hoped. The expected cause is that the stop/start behaviour it seeks to avoid is caused in real life by elements such as pedestrians crossing the road, cyclists, parked cars and speed bumps, which are not modelled in simulation. The study would suggest that journey time is the most important component followed by used capacity cost and fuel consumption and emissions cost. This is supported by the earlier estimation that weight settings of 75%,15% and 10% for journey time cost, used capacity cost and fuel consumption and emissions cost would give the variant of the TraffCon solution that performs best.

7.2.7 The effect of reducing the penetration of the solution

Test Scenario

An important consideration for any vehicular communications based traffic management solution is how the performance is affected as the percentage of all vehicles in the road network that run the solution is reduced. In previous tests where the TraffCon solution was considered it was assumed all vehicles ran the TraffCon solution. In order to evaluate the efficiency of the proposed TraffCon solution as the level of its penetration varies the following experiment was performed examining three different scenarios, which involve TraffCon and two competing approaches. In order to reduce the influence of noise in the results, the experiments were run three times using different seeds and the results were averaged.

Case (1): before each vehicle embarks on its journey it selects a shortest route using the A* shortest path algorithm [63]. The vehicle does not deviate from this route. The shortest path algorithm weights the road segments along a path according to the estimated time taken to traverse them. This is calculated by multiplying the length of the segment by the speed limit on that segment and adding a time penalty based on the intersection type found on the segment. This solution is referred to as A* based on distance hereafter. The penetration of A* based on distance is 100%.

Case (2): before each vehicle embarks on its journey it selects a shortest route using the A* shortest path algorithm. In this instance however the shortest path algorithm weights the road segments along a path using the actual time taken to traverse them as measured by vehicles travelling on the road network. These time values are gathered in the same manner as for the TraffCon solution. This solution is referred to as A* based on time hereafter. The penetration of A* based on time is 100%.

Case (3):each vehicle drives to its own destination according to the route management solution with dynamic adaptation during the journey. The weights of the fitness function are set at w_1 = .7 , w_2 = .2 and w_3 = .1 and k was set to 20. The level of penetration of the TraffCon solution is varied.

The simulation time was set at two hours. In each simulation, the number of vehicles on the road was varied and average journey time and fuel economy was measured.The origin-destination pairs are kept constant i.e. the same vehicles are attempting to complete the same journeys in all cases. Border behaviour is not an issue as journeys occur within the simulation area. Whenever a journey is completed, another vehicle commences a journey, so the number of vehicles is maintained constant during the whole simulation duration.

Results and Analysis

The results for average journey time are compiled in figure 7.27. The graph shows results for A* based on both distance and time and TraffCon at selected penetrations levels, in the graph the penetration level is expressed as a decimal fraction. The selected penetration levels shown are 90%, 70%, 50% and 10%. The behaviour of the TraffCon solution as the penetration level is reduced is difficult to predict. This is because vehicles running the solution will capture the effect of other vehicles on the road network when they measure time etc. and compensate forming some kind of balance with what vehicles not running the solution are doing. With less vehicles collecting data the information about the state of the road network will not be updated as frequently, this may also have an unpredictable effect. This unpredictability can be seen in the graph of journey time against number of vehicles as there is not a clear demarcation between all the variants of TraffCon running at a reduced penetration.

All solutions are outperformed by the TraffCon variant that has 100% penetration. The variants at 90% and 70% still out perform the A* based on time approach (used for comparison in section 7.2.2) as their values for journey time remain below those for A* based on time throughout. The variant of TraffCon at 50% penetration out performs A* based on time for large portions of the graph but gives journey times that are marginally higher or approximately the same as A* based on time in the range of 350-450 vehicles. The variant of TraffCon performs approximately the same as A* based on time until th number of vehicles reaches 550, then it performs worse until 650 vehicles from where it gives lower journey times for the remainder of the graph.

In the first portion of the graph all solutions behave similarly as there are a very small number of vehicles present on the road network and traffic congestion is not an issue. As the number of vehicles increases average journey time begins to increase for all solutions. As the road network becomes saturated with vehicles, journey times begin to rise steeply. In figure 7.28 a snapshot of the solutions is taken when congestion is becoming a serious

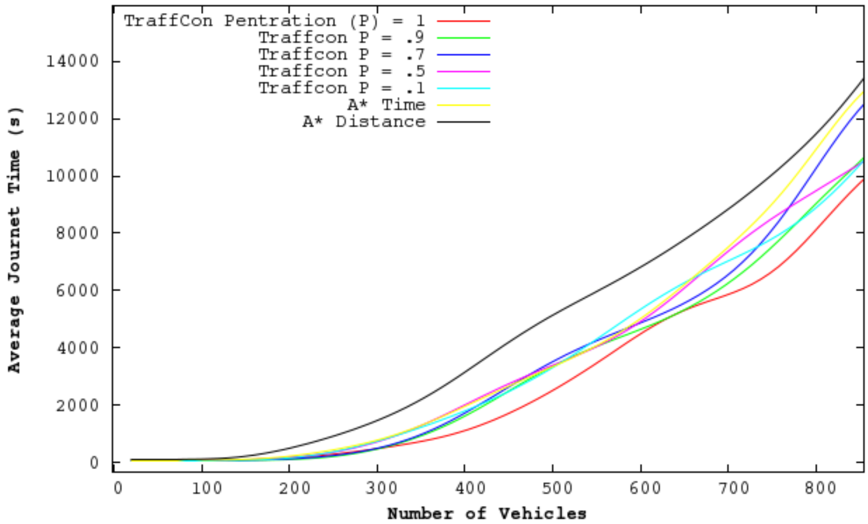


Figure 7.27: Average journey time against number of vehicles

issue at the 600 vehicle mark. When compared with A* based on time 100% penetration TraffCon gives an average journey time 12.7% lower, 90% penetration TraffCon gives an average journey time 8.8% lower, 70% penetration TraffCon gives an average journey time 4.9% lower, 50% penetration TraffCon gives an average journey time 3.1% lower and 10% penetration TraffCon gives an average journey time that is 6.8% higher.

While the variant of the TraffCon solution at 90% penetration cannot be said to perform better than that at 70%, 70% better than 50% and 50% better than 10% for all numbers of vehicles, on average the higher the penetration level the better the TraffCon Solution performs. The TraffCon solution outperforms A* based on time with the penetration level as low as 50% and gives results that are comparable to A* based on time with the penetration level as low as 10%. The 10% variant of TraffCon gives broadly similar but some times higher journey times to A* based on time (higher by as much as 6.8%) however, given that A* has a penetration of 100% and not 10% they still compare favourably.

The results for average fuel economy are compiled in figure 7.29. Once again the graph shows results for A* based on both distance and time and TraffCon at selected penetration levels of 90%, 70%, 50% and 10%.In this graph of average fuel economy against number

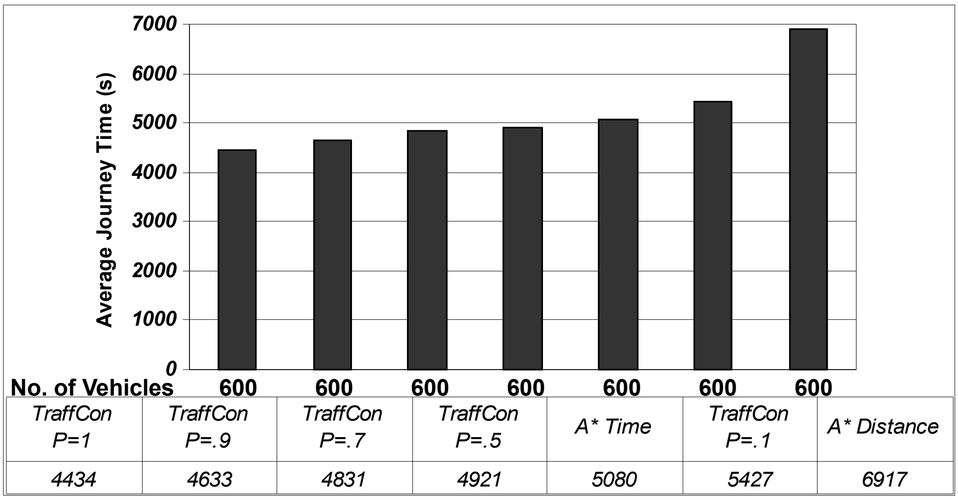


Figure 7.28: Average journey time when number of vehicles is 600

of vehicles there is a clear demarcation between all solutions. The variant of TraffCon with a 100% level of penetration performs better than A* based on time. The variant of TraffCon with 90% penetration performs slightly worse than A* based on time, the 70% variant performs worse than the 90% variant, the 50% worse than the 70% and the the 10% worse than the 50%.

In conclusion the performance of the TraffCon solution is resilient to a reduction in the level of its penetration where journey time is considered. When fuel economy is considered however performance is quickly affected and becomes worse than A* based on time at a penetration level of 90%. Why is this? In the previous section it was seen what an important contribution the capacity component of the fitness function plays in improving fuel economy. The measurment of used capacity is most affected by the reduction in penetration level because if a vehicle is not reporting its whereabouts to the server, then its contribution to the used capacity cannot be measured. This inaccuracy in the used capacity measurement affects the accuracy of the used capacity component of the fitness function and consequently the fuel economy performance. If journey time is the only consideration then a penetration as low as 50% may be exceptable as the solution is still beneficial

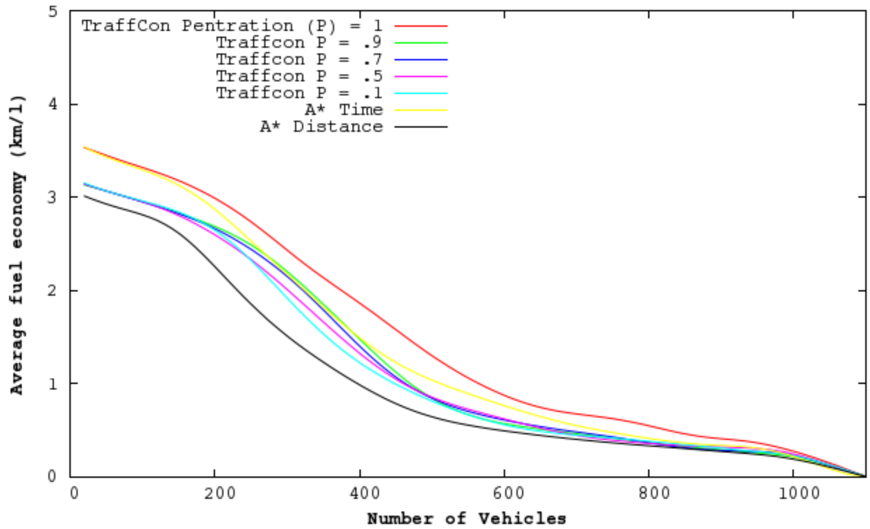


Figure 7.29: Average fuel economy against number of vehicles

and would cost significantly less than a penetration of 100%. If however fuel econmy is considered to be an important metric then the penetration level must be kept high or the ability of vehicles to measure used capacity as the percentage penetration level drops, must be improved. If for example vehicles employing the TraffCon solution could sense nearby vehicles not using TraffCon and count their contribution to used capacity then this issue could be overcome.

7.2.8 Investigation into global improvements versus individual improvements

The aim of this section is to investigate if the global performance benefits of the TraffCon solution as shown in section 7.2.2 come at the expense of certain individuals. In order to determine this the following metrics; average, best and worst individual speed and fuel economies are examined. These metrics were measured in the experiment described in section 7.2.5. The best and worst individual cases are used to explore whether global improvements are achieved at the expense of individual drivers. Solutions where global improvements are achieved but certain individuals are harshly punished would not be desirable. For the purposes of this investigation results obtained from the Chicago map will be

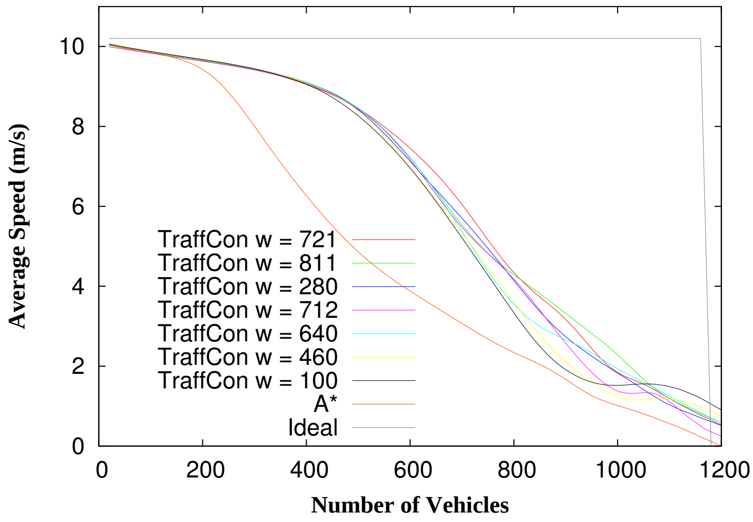


Figure 7.30: Average speed against number of vehicles for Chicago map

used. The figures to be displayed were compiled in the same fashion as previously outlined for figure 7.24. Any observations made here after could similarly have been made using results from the Boston Map.

Figure 7.30 contains a graph of average speed against number of vehicles for the Chicago map. It was discerned in the previous section that the best performing variant of the TraffCon solution for the Chicago Map was that with weights set to w_1 = 0.7, w_2 = 0.2 and w_3 = 0.1. It can be clearly seen in figure 7.30 that this solution has the highest results for average speed. It is clear that TraffCon solutions generate performance improvements by increasing the average speed of vehicles. The correlation between average speed and completed journeys is highlighted more clearly in figure 7.31. In this diagram corresponding portion of the graphs between 500 and a 1000 vehicles for figures 7.24 and 7.30 are expanded. The diagram illustrates the fact when a solution is seen to be the best in terms of completed journeys it also has the highest average speed. A similar trend can be seen if we look at results for average fuel economy as shown in figure 7.32.

The average speed and fuel economy results can show us how global improvements are made when all vehicles travelling in the road network are considered but does not indicate how individual vehicles may be affected. To uncover the effect on individual drivers the best and worst individual speed and fuel economies are examined. Firstly looking at the best

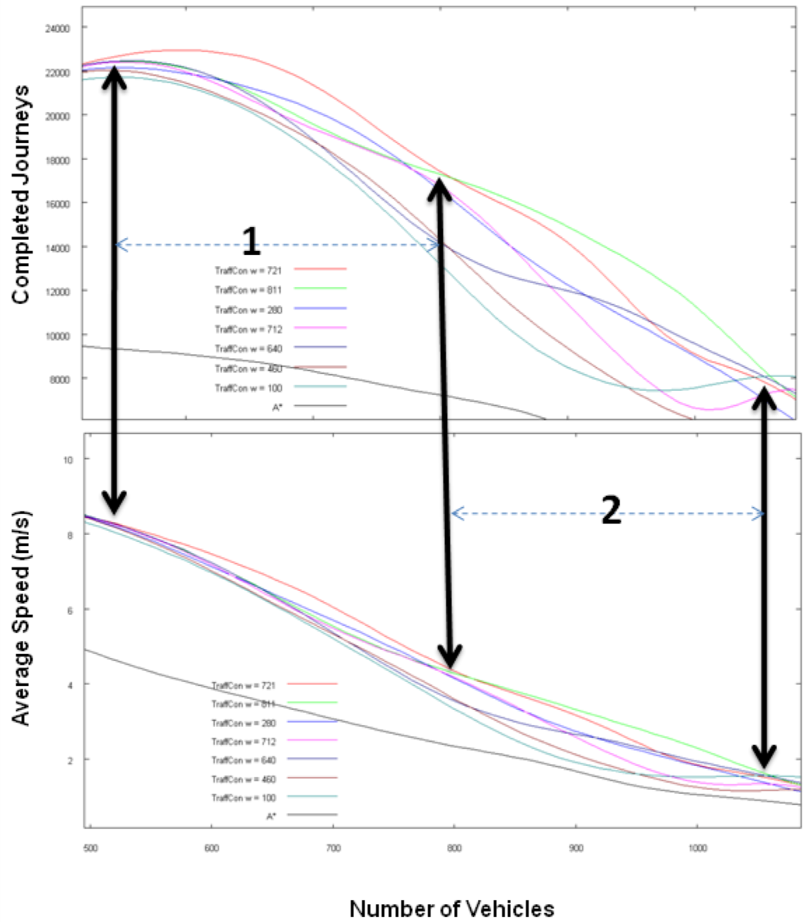


Figure 7.31: The correlation between average speed and completed journeys

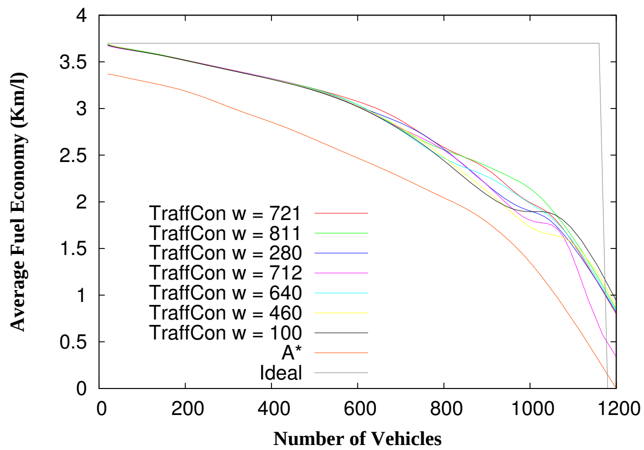


Figure 7.32: Average fuel economy against number of vehicles for Chicago map

performing individual where speed and fuel economy are concerned as shown in figures 7.33 and 7.34. From these graphs it can be seen that when comparing TraffCon Solutions with the baseline A* shortest path solution that one of the contributing factors to the overall improvements is that there is an improvement in the best individual performances.

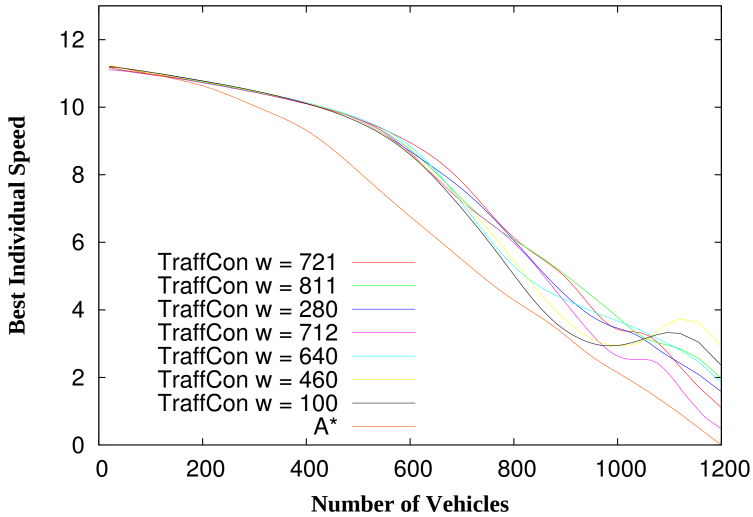


Figure 7.33: Best individual speed against number of vehicles for Chicago map

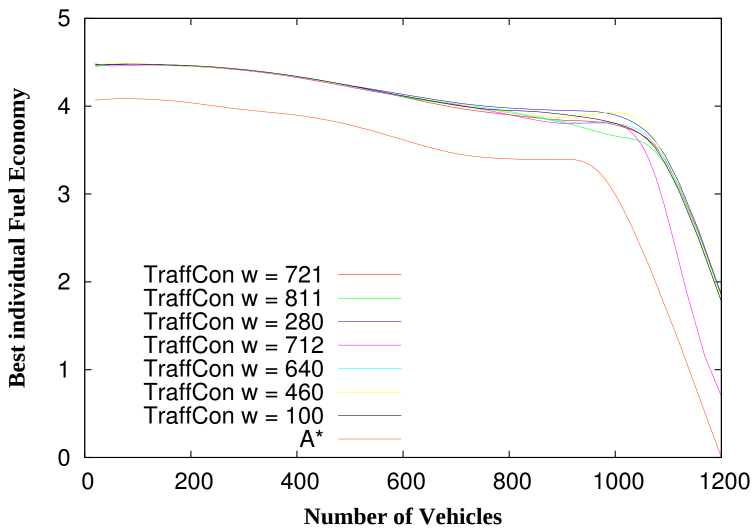


Figure 7.34: Best individual fuel economy against number of vehicles for Chicago map

Next examining the worst individual performances where speed and fuel economy are concerned as shown in figures 7.35 and 7.36. From these graphs it can be seen that when comparing TraffCon Solutions with the baseline A* shortest path solution that one of the contributing factors to the global improvement is that there is an improvement in the worst individual performances.

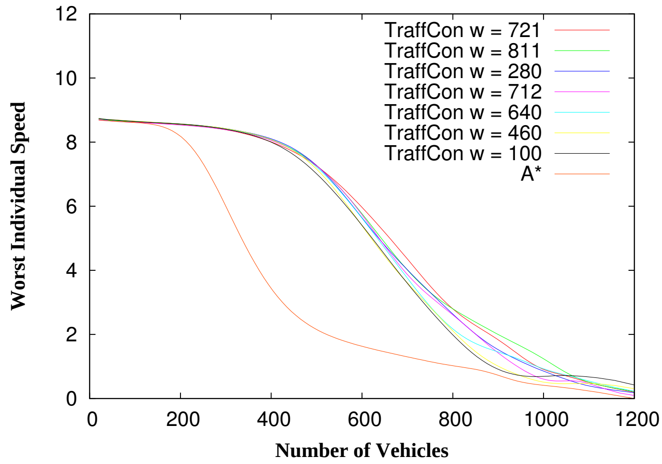


Figure 7.35: Worst individual speed against number of vehicles for Chicago map

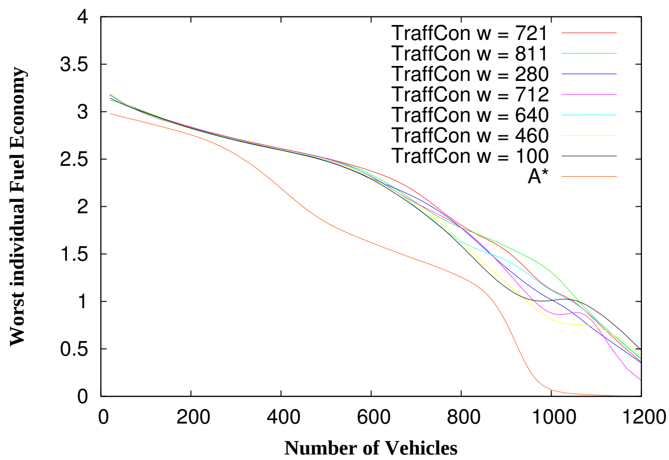


Figure 7.36: Worst individual fuel economy against number of vehicles for Chicago map

In conclusion, we have seen that the TraffCon solution has global performance benefits for vehicles travelling in a road network over existing approaches. These improvements in performance come in the form of shorter journey times/higher average speeds and improved fuel economy. There was a concern that these improvements may have been made by dramatically improving the performance of some vehicles while a proportion of vehicles experience a performance drop. In this section we have seen that the global benefits did not come at the expense of certain individuals but rather by increasing the average speed and fuel economy of all vehicles.

7.2.9 Summary of Results

After analysis of results from testing of the best route selection algorithm the following has been shown. The proposed solution reduces journey times, fuel consumption, fuel emissions and fuel costs. The solution brings improvements whether operated in a non-adaptive or adaptive fashion but solutions that use adaptation perform better. Performance of the algorithm increases as its k parameter increases but the majority of the increase occurs in the early increments as seen in a Pareto curve. Consequently in practice k may be set at a relatively low value approximately 30 to achieve good performance. The algorithm performs well irrespective of the road map used e.g. an irregular road network or a regular "Manhattan grid". All three cost components of the algorithm's fitness function contribute to finding the best performing variant of the TraffCon solution. In practice for best performance of the TraffCon solution the three cost components; Journey Time Cost, Used Capacity Cost and Fuel Consumption and Emissions Cost should be assigned weighting factors of 0.75, 0.15 and 0.1 respectively. Finally the global performance improvements achieved by the TraffCon solution do not come at the expense of individuals. This means that when using the TraffCon solution there is no concern that certain individuals will be harshly punished by the solution despite the overall improvement in performance.

7.3 Prioritised Data Exchange - Testing

The prioritized data exchange algorithm was outlined in the previous chapter. The algorithm initiates and manages the exchange of data between a vehicle and roadside units it passes when travelling through a sparse roadside-vehicle communications network. The results from the analysis of the prioritised data exchange algorithm are presented in sections 7.2.3 and 7.2.5. The evaluation of the algorithm is divided into two stages. In these stages the following aspects are examined in order to determine the overall effectiveness of the algorithm:

- The levels of data which can be exchanged
- The implications of data prioritisation

The experiments are carried out using simulation based testing. The test setup is described in the next section.

7.3.1 Test Setup

In order to evaluate the prioritized data exchange algorithm simulations were performed with TraNS [128, 127] (Traffic and Network Simulation Environment). TraNS uses the Simulation of Urban MObility (SUMO) traffic simulator to provide realistic modelling of vehicular mobility. The road network information can be taken from real world road map sources such as the US Census Bureau's TIGER/Line files [8]. The ns-2 network simulator is used to provide simulation of wireless communications. Support for realistic IEEE 802.11p is possible using ns version 2.33 which has overhauled the modelling and simulation of IEEE 802.11 for this purpose as outlined in [32].

To model a sparse roadside-vehicle communications network an infrastructure point is placed at a signalised intersection in the centre of a sub-network of the road network of Chicago, IL as shown in figure 7.37. Vehicles drive by the infrastructure point and exchange data according to the prioritized data exchange algorithm. Realistic vehicular traffic patterns are generated using TraNS. The access technology used is IEEE 802.11p.

[8]The US Census Bureau's TIGER/Line files:http://www.census.gov/geo/www/tiger/

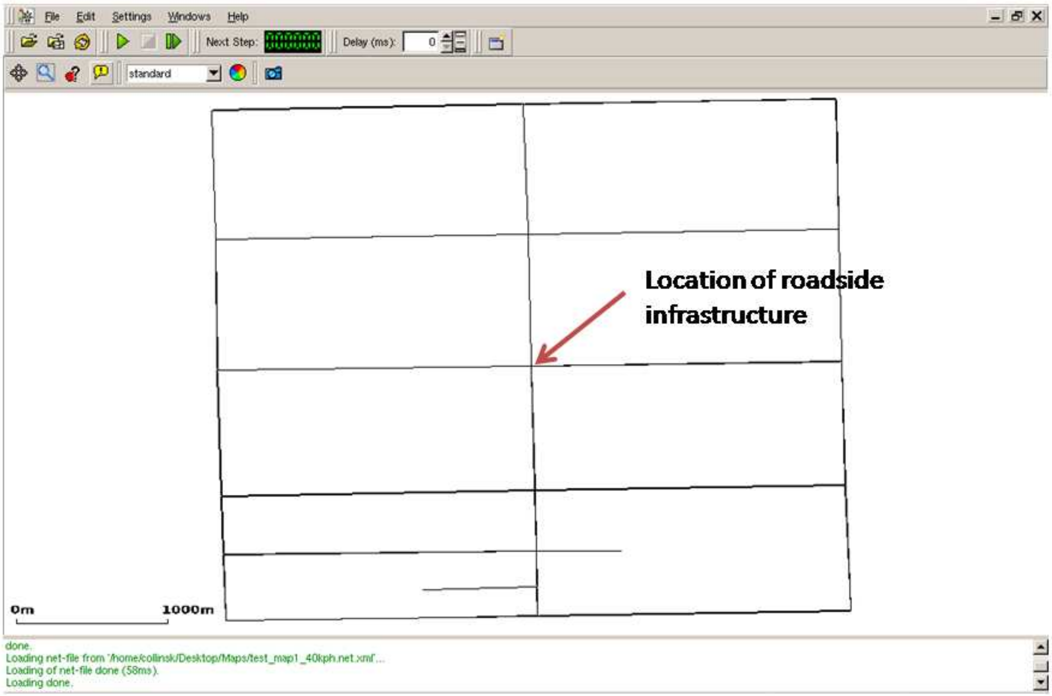


Figure 7.37: Sub network of Chicago road network visualised in SUMO with infrastructure point indicated

The detail of IEEE 802.11p ns-2 settings for simulation are explained in section 7.3.1.1 and the individual test scenarios are outlined separately and in detail for each experiment in the following sections. There is analysis of the test results for each experiment and finally the overall conclusions which can be derived from these results is summed up in the summary of results section.

7.3.1.1 IEEE 802.11p ns-2 settings for Simulation

When performing wireless network simulation using the ns-2 simulator (version 2.34) there are a variety of parameters which must be set to mimic a particular technology. In the case of IEEE 802.11 networks, MAC and physical layer settings must be tweaked to indicate a particular flavour of 802.11 for example a, b, g or in this case p. The settings for 802.11p are presented in detail hereafter.

MAC Settings

Setting	Value	Note
CWMin_	15	Congestion Window Minimum
CWMax_	1023	Congestion Window Maximum
SlotTime_	0.000013	Fast slot time (s)
SIFS_	0.000032	Short Interframe Space, time interval (s)
ShortRetryLimit_	7	Max number of retransmission attempts of an RTS or data packet when RTS/CTS is not used
LongRetryLimit_	4	Max number of retransmission attempts of a data packet when RTS/CTS is used
RTSThreshold_	2346	Point at which the data packet size is too small to initiate the RTS/CTS function. Having a high value like 2346 essentially switches RTS/CTS off.
HeaderDuration_	0.000040	PLCP Preamble + Header = 120 bits (192bits + 48bits). BPSK modulation plus $^1/_2$ coding rate implies 6Mbps, giving header duration of 40s
SymbolDuration_	0.000008	For a 20MHz channel, each periodic waveform/symbol (plus a guard interval to prevent inter-symbol interference) takes 8s to send.
BasicModulationScheme_	0	Code to indicate modulation scheme 0 is BPSK with coding rate $^1/_2$ & is the basic modulation scheme for header & control packets, results in data rate of 6 Mbs[1].

Table 7.4: MAC settings

As indicated in the notes above these parameters may need to be tweaked for a variety of reasons. One example is a situation where the 802.11 multiple data rates mechanism is used and data rates vary depending on the distance between node and access point. In this instance the modulation scheme parameter would need to be varied accordingly.

[1] The modulation scheme may be varied for every individual MAC frame. This is done by flagging a variable in the frame common header e.g. the modulation scheme could be changed to QAM64 and $^3/_4$coding rate (54Mbs). Used to implement multiple data rates where necessary

Wireless Physical Settings

Setting	Value	Note
CSThresh_	3.9810717055349694e-13	Carrier Sense Threshold (Watts) - Depends on wireless interface sensitivity of chipset. Set at -94dBm in this example
noise_floor_	1.26e-13	The Noise Floor (Watts) - the measure of the signal created from the sum of all the noise sources and unwanted signals within a system. Again depends on the chipset, set at -99dBm in this example
PowerMonitorThresh_	3.981071705534985e-18	Power Monitor Threshold (Watts) this example shows a power monitoring sensitivity of -174dBm
Pt_	0.1	Transmitted Signal Power (Watts) available transmit power settings will come from chipset
freq_	5.9e+9	The frequency band- in this case 5.9GHz
L_	1.0	System Loss Factor (1.0 is default radio circuit gain/loss
HeaderDuration_	0.000040	40s see MAC setting
PreambleCaptureSwitch_[1]	1	Preamble Capture on or off (0/1)
DataCaptureSwitch_	1	Data Capture on or off (0/1)
SINR_PreambleCapture_	3.1623	Rise in signal strength (Watts) required to activate preamble capture in this case 5dB
SINR_DataCapture_	10.0	Rise in signal strength (Watts) required to activate preamble capture in this case 10dB
trace_dist_	1e6	ns-2.34 traces message drop events at the physical layer. The *trace_dist_* parameter is a distance in metres between sender and receiver beyond which drops related to power will not be traced. The distance shown of 1000km is effectively infinity.

Table 7.5: Wireless physical settings

As indicated in the notes above these parameters may need to be adjusted for a number of reasons. For example whether or not preamble capture is in use must be indicated by turning the preamble capture switch on or off. It is also important that when a final decision is made on the wireless chip-set that simulation parameters such as the carrier sense threshold mirror the characteristics of the chipset.

RF Propagation Settings

Aside from MAC and physical layer settings ns-2 also allows the setting of an RF Propagation model to represent the environment in which the wireless communications take place.

[1] The capture capability is a very important feature in this model. It allows a receiver to distinguish the MAC frame header and body, using different criterion to processing them. MAC header is transmitted with defined BPSK modulation, while the MAC data can be coded in a much higher modulation scheme. A real IEEE802.11 product can pick up a frame with stronger header signal among multiple frames and complete its data reception. Such a technology, called capture, is wildly used and can prove very effective in enhancing reception probabilities. NS-2 users can turn on a TCL switch to simulate this feature or turn off the switch to simulate original style implementation without capture in version 2.34.

These models are used to predict the received signal power of each packet. At the physical layer of each wireless node, there is a receiving threshold. When a packet is received, if its signal power is below the receiving threshold, it is marked as error and dropped by the MAC layer. In earlier versions of ns-2 three models were available - Free-space, Two-ray ground reflection, and Shadowing[1]. The latest version of ns-2 (version 2.34) adds support for the Nakagami propagation model.

Compared to the existing models (shadowing and two-ray ground), the Nakagami RF model has more configurable parameters to allow a closer representation of the wireless communication channel. It is able to model from a perfect free space channel, to a moderate fading channel e.g. on a highway, or even to a dramatically fading channel such as found in urban environments. A Nakagami RF propagation model representing a typical radio propagation channel as found on a highway was used for ns-2 simulations discussed in this thesis[2].

7.3.2 Investigation into levels of data which can be exchanged

In order to determine the limits of the algorithm it was tested for the worst case scenario i.e. large numbers of cars present, all of which are trying to receive x amount of data from the roadside infrastructure. Three volumes of vehicular traffic are examined moderately heavy, heavy and very heavy traffic. For moderately heavy traffic vehicles are flown toward the intersection where the infrastructure point is located from all four adjoining roads at a rate of one every 8 seconds for 200 seconds. For heavy traffic the rate is one every 4 seconds and for very heavy the rate is one every 2.6667 seconds. In each case the simulation is allowed continue until all the traffic has passed through the intersection and is out of range of the infrastructure point. This results in 100 vehicles, 200 vehicles and 300 vehicles passing through the intersection in the moderately heavy, heavy and very heavy cases respectively.

For each of the three traffic patterns measurements are taken for three different cases:

[1] More detail can be found in the ns manual Chapter 18: Radio Propagation Models

[2] For further detail on the Nakagami Model being used see section 1.3.5 of Overhaul of IEEE 802.11 Modeling and Simulation in NS-2 (802.11Ext):
http://dsn.tm.uni-karlsruhe.de/medien/downloads_old/Documentation-NS-2-80211Ext-2008-02-22.pdf

Case (1): All vehicles try to receive 250kB of data from the roadside infrastructure.

Case (2): All vehicles try to receive 500kB of data from the roadside infrastructure.

Case (3): All vehicles try to receive 1000kB of data from the roadside infrastructure.

In the simulations the infrastructure informs nearby vehicles of its presence with a beacon, which is periodically broadcast. Vehicles respond to the beacon with a service request if they have not already received it. In the previous section the traffic management service powered by the best route selection was evaluated. In Chapter 6 the data that must be gathered by this service was summarised in table 6.2. In table 7.6 the data size of this information is summarised, the total size of data gathered is 56 bytes if there is an access point at every intersection. If the density of access points were less than one at every intersection then multiple road segments may be passed between access points, then the size of data gathered would be the number of road segments times 56 bytes. The size of the service request is set at 10kB to allow for the vehicles running the traffic management service and many additional services or for a drop in access point density leading to an increase in the amount of gathered data that has to be sent.

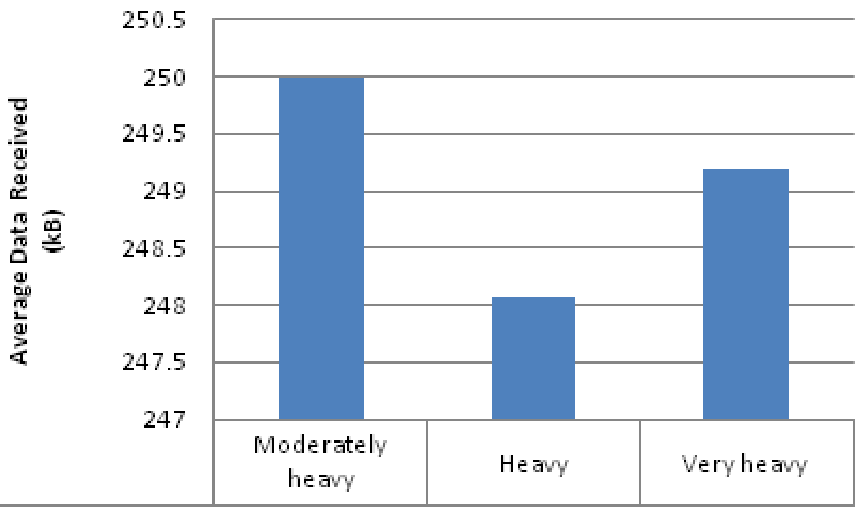
Data Field	**Variables Required**	**Data Size (bytes)**
Vehicle ID	8 characters	8
Time Stamp	date and time	8
Last Segment ID	4 floating point numbers	16
Segment Time (ms)	long integer	4
Next Segment ID	4 floating point numbers	16
Fuel Consumption (ml)	floating point number	4

Table 7.6: Size of data gathered by vehicles

Once the service request is received, the infrastructure point then transmits data to the vehicle. In the case of the traffic management service, the service returns the list of road segments a vehicle must take on route to its destination. As seen in table 7.6 road segment IDs are represented by four floating point numbers or 16 bytes of data. So given a journey requiring 20 road segments be crossed 320 bytes of data are required. In the test the amounts of data sent are set much higher, 250kB, 500kB and 1000kB. It was found that the data requirements of the traffic management service did not tax the network so the data amounts were increased to mimic the vehicles running additional services. To this was done to see if the infrastructure could support the vehicles running the traffic management service and additional services with higher data requirements. Data is transmitted at a rate of 6Mbps, this is the lowest rate available in the IEEE 802.11p standard, which offers data rates ranging from 6-27 Mbps. UDP is used at the transport layer and the access technology used is IEEE 802.11p. The ns-2 MAC and Phy settings used for IEEE 802.11p are explained in detail in 7.3.1.1, where modelling of RF propagation is also discussed. Further information on mobility settings and a sample TCL file for 802.11p settings can be found in Appendix C. In each case the average data received and data received and lost is measured for each vehicle.

7.3.2.1 Case (1) 250kB of data transmitted per vehicle

In figure 7.38 the average data received by each vehicle is shown for the case where vehicles are expected to receive 250kB of data. It can be seen that there is no data lost for the scenario with moderately heavy traffic as the average is 250kB. The average data received for the heavy traffic and very heavy traffic cases is 248.1kB and 249.2kB respectively. This means that the average amount of data received in those cases is 99.2% and 99.7% respectively.



	Moderately heavy	Heavy	Very heavy
Average Data Received (kB)	250	248.065	249.19

Figure 7.38: Average Data received by each vehicle

The data received by each vehicle for the moderately heavy, heavy and very heavy traffic cases is shown in figures 7.39, 7.40 and 7.41

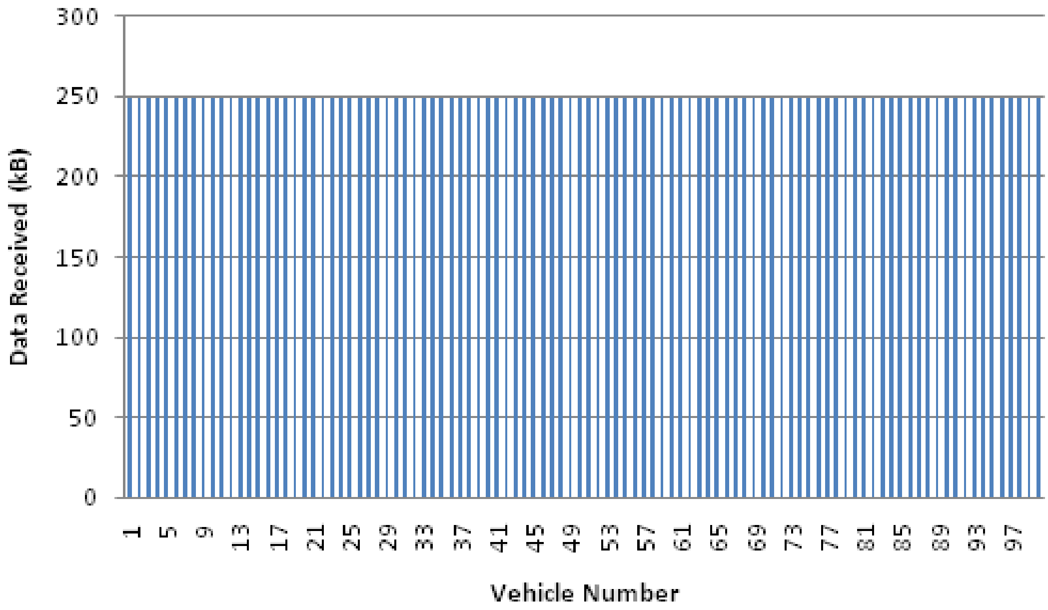


Figure 7.39: Data received by each vehicle when traffic is moderately heavy

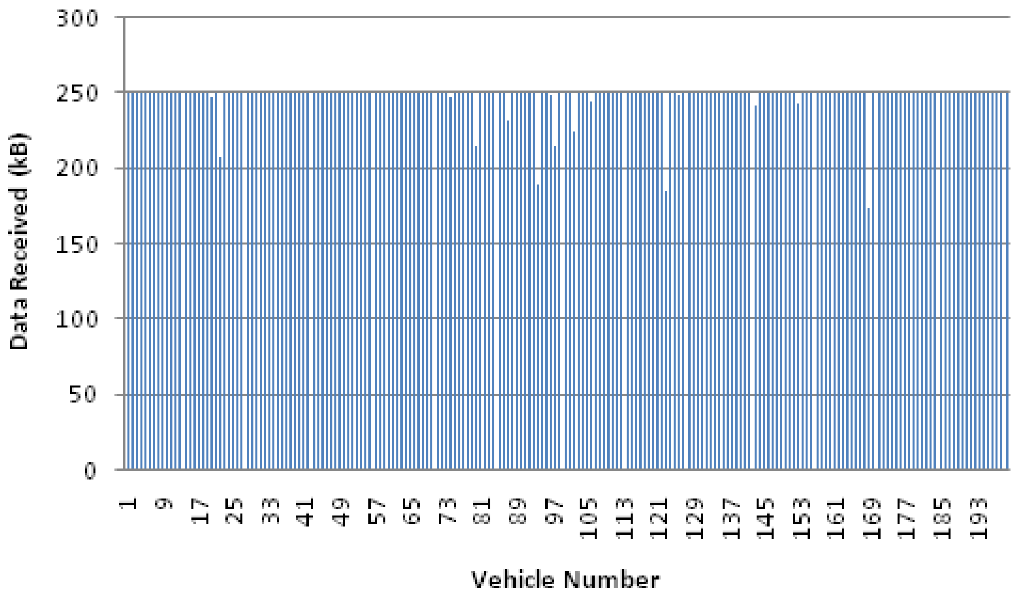


Figure 7.40: Data received by each vehicle when traffic is heavy

It can be clearly seen where individual vehicles lost data, and how much data was lost

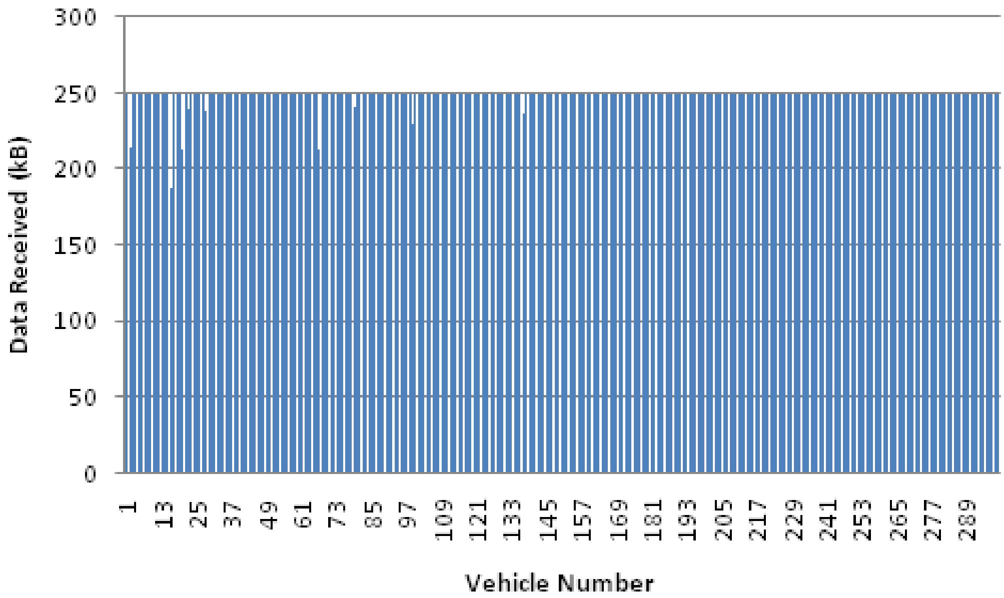


Figure 7.41: Data received by each vehicle when traffic is very heavy

in the heavy and very heavy traffic cases in figure 7.42 and 7.43

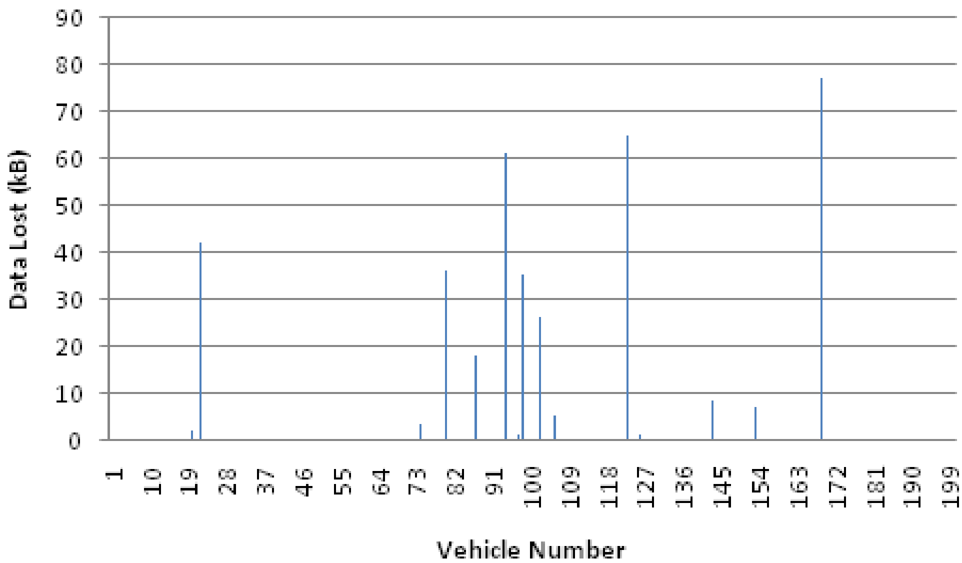


Figure 7.42: Data loss for each vehicle when traffic is heavy

The percentage of vehicles that receive the full amount of data expected is shown in figure 7.44. Again it is highlighted that 100% of vehicles received the expected amount of

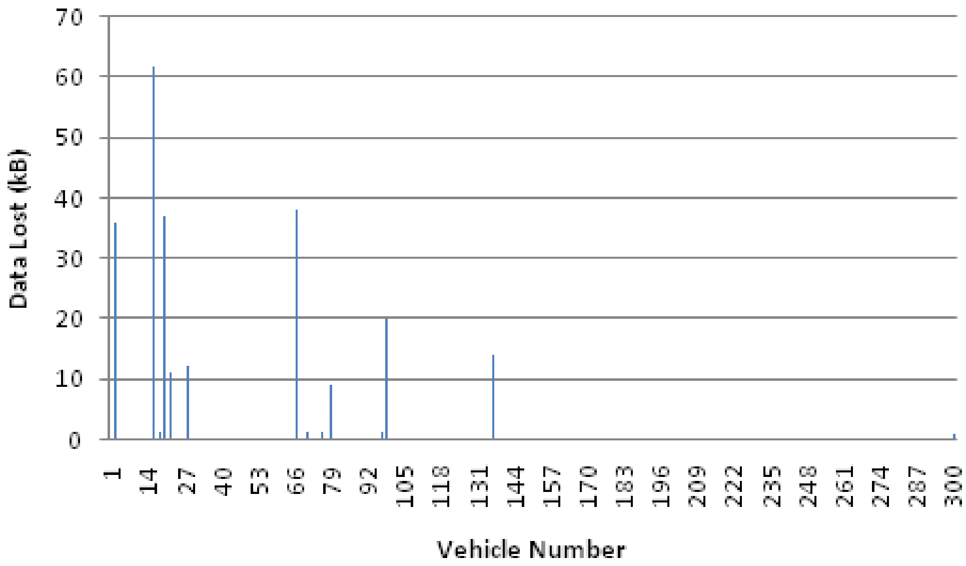


Figure 7.43: Data loss for each vehicle when traffic is very heavy

data in the scenario with moderately heavy traffic. In the heavy and very heavy traffic cases 92.5% and 95% of vehicles respectively received the expected amount of data.

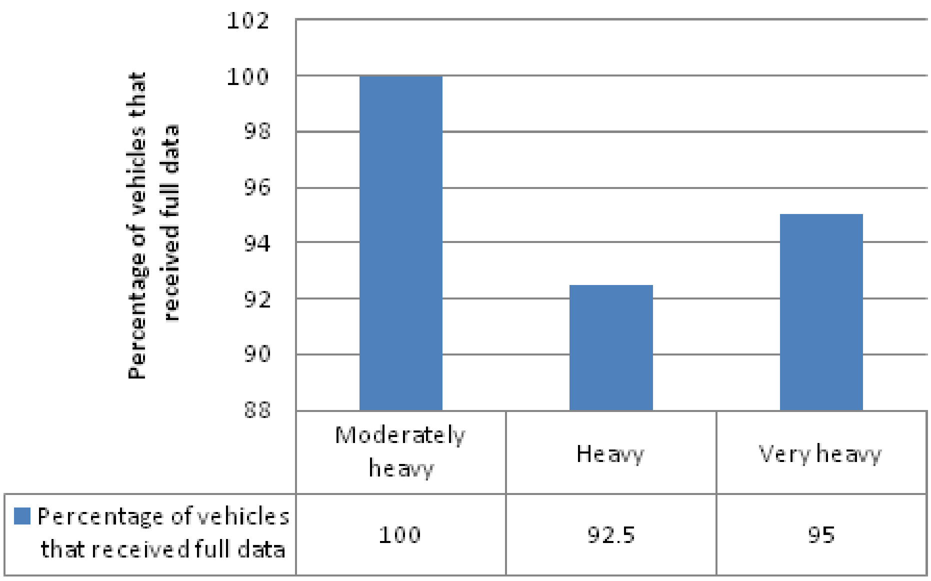


Figure 7.44: Percentage of vehicles that receive full amount of data

7.3.2.2 Case (2) 500kB of data transmitted per vehicle

In figure 7.45 the average data received by each vehicle is shown for the case where vehicles are expected to receive 500kB of data. The average data received for the moderately heavy traffic, heavy traffic and very heavy traffic cases is 497.53kB, 475kB and 488.2kB respectively. That is 99.5%, 95% and 97.6% of the expected amount in each case.

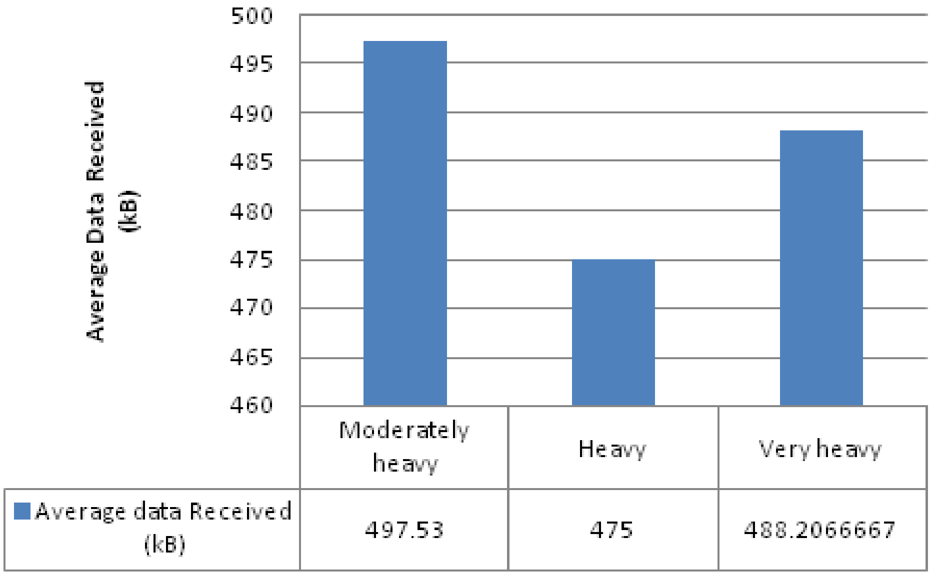


Figure 7.45: Average Data received by each vehicle

The data received by each vehicle for the moderately heavy, heavy and very heavy traffic cases when the expected amount is 500kB is shown in figures 7.46, 7.47 and 7.48

Instances where individual vehicles lost data, and how much data was lost in the moderately heavy , heavy and very heavy traffic cases are shown in figures 7.49, 7.50 and 7.51.

The percentage of vehicles that receive the full amount of data expected, when the expected amount is 500kB is shown in figure 7.52. In the moderately heavy, heavy and very heavy traffic cases 91%, 70.5% and 79.3% of vehicles respectively received the expected amount of data.

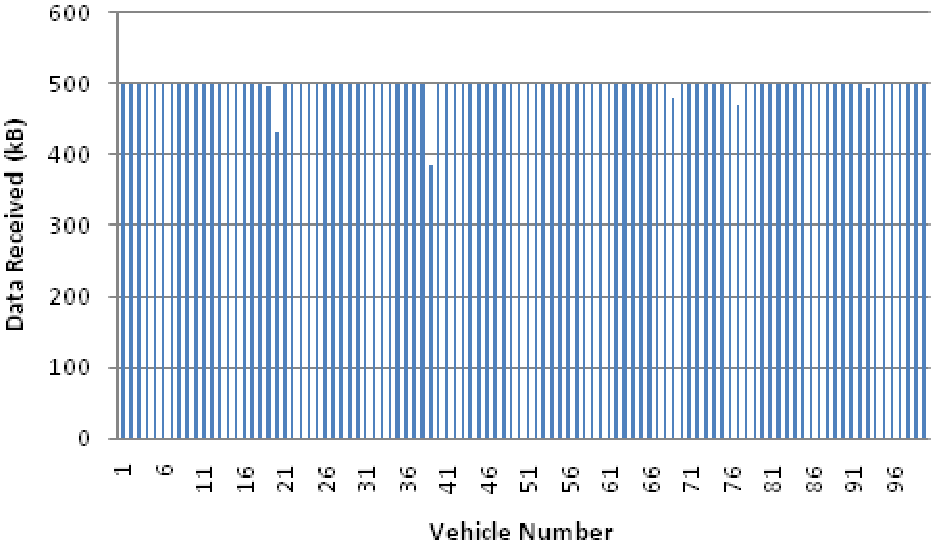


Figure 7.46: Data received by each vehicle when traffic is moderately heavy

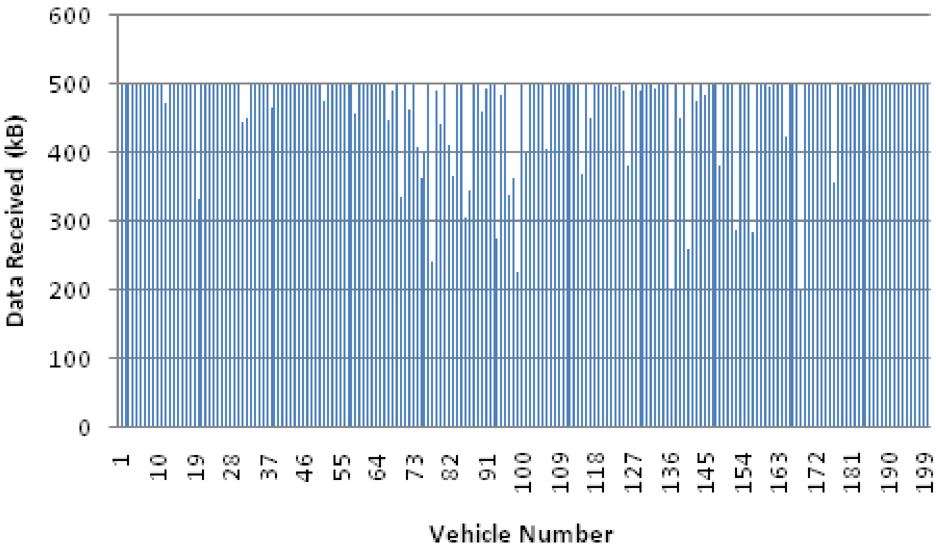


Figure 7.47: Data received by each vehicle when traffic is heavy

7.3.2.3 Case (3) 1000kB of data transmitted per vehicle

In figure 7.53 the average data received by each vehicle is shown for the case where vehicles are expected to receive 1000kB of data. The average data received for the moderately heavy traffic, heavy traffic and very heavy traffic cases is 972.16kB, 799.12kB and 815.11kB

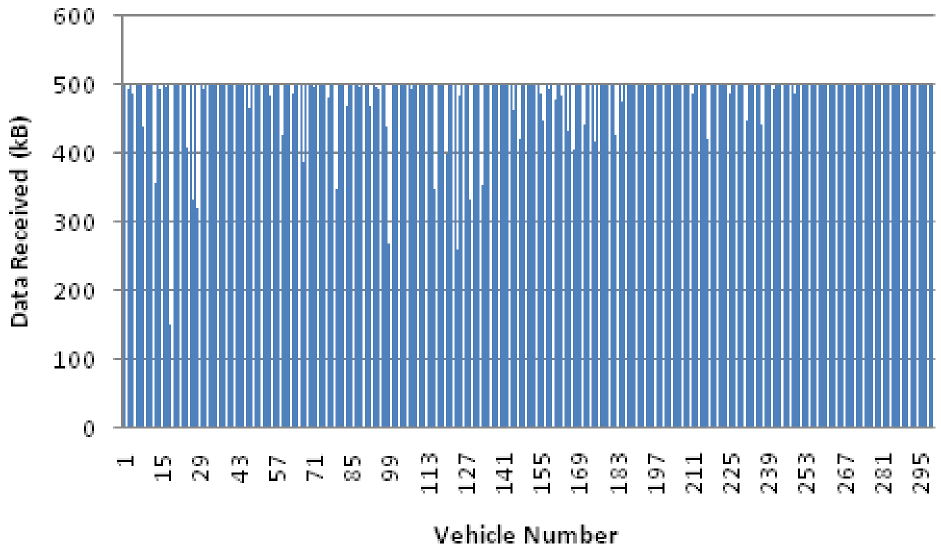


Figure 7.48: Data received by each vehicle when traffic is very heavy

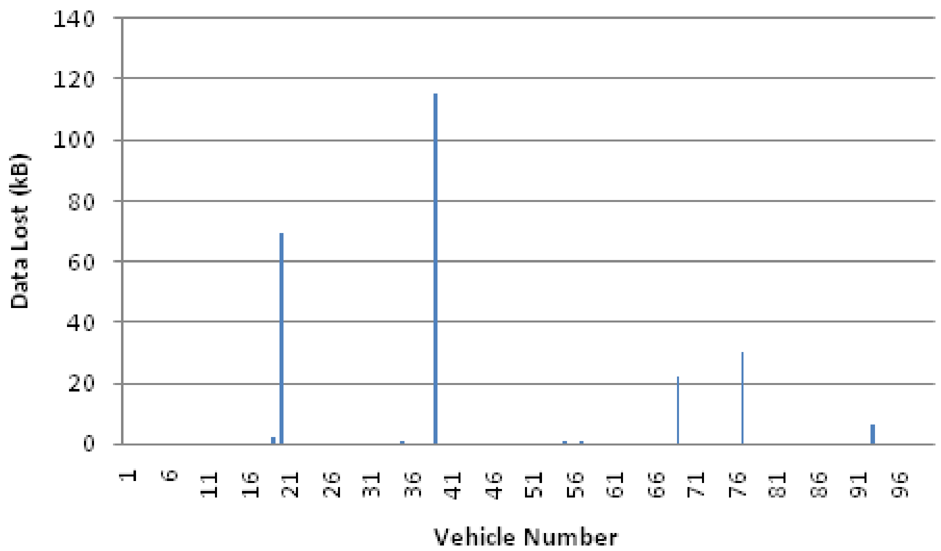


Figure 7.49: Data loss for each vehicle when traffic is moderately heavy

respectively. That is 97.2%, 79.9% and 81.5% of the expected amount in each case.

The data received by each vehicle for the moderately heavy, heavy and very heavy traffic cases when the expected amount is 1000kB is shown in figures 7.54, 7.55 and 7.56

Instances where individual vehicles lost data, and how much data was lost in the mod-

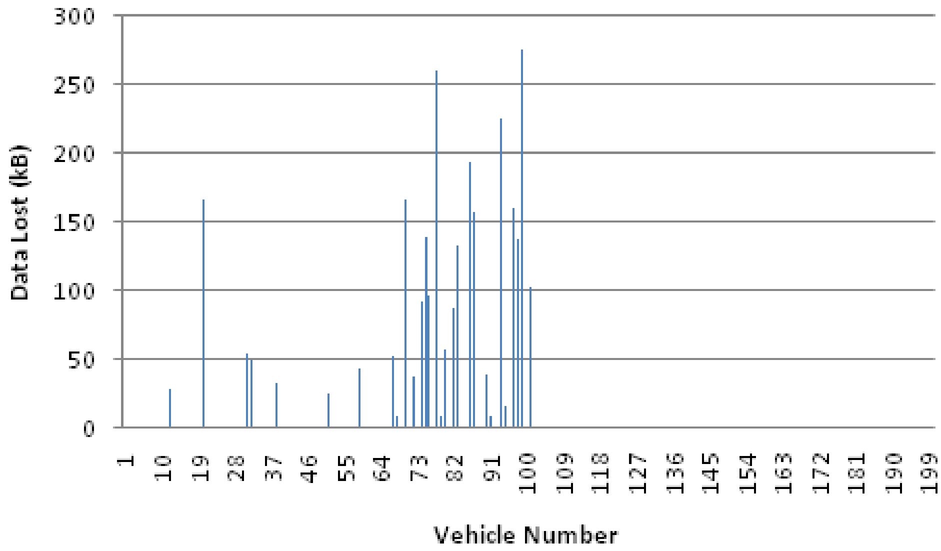


Figure 7.50: Data loss for each vehicle when traffic is heavy

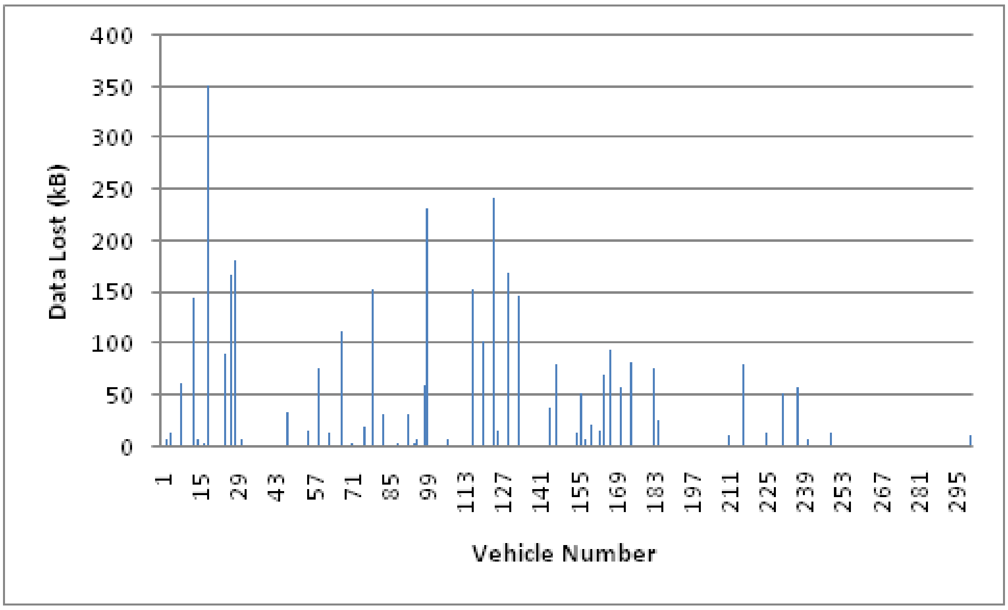


Figure 7.51: Data loss for each vehicle when traffic is very heavy

erately heavy , heavy and very heavy traffic cases are shown in figures 7.57, 7.58 and 7.59.

The percentage of vehicles that receive the full amount of data expected, when the expected amount is 1000kB is shown in figure 7.60. In the moderately heavy, heavy and very heavy traffic cases 68%, 38.5% and 38% of vehicles respectively received the expected

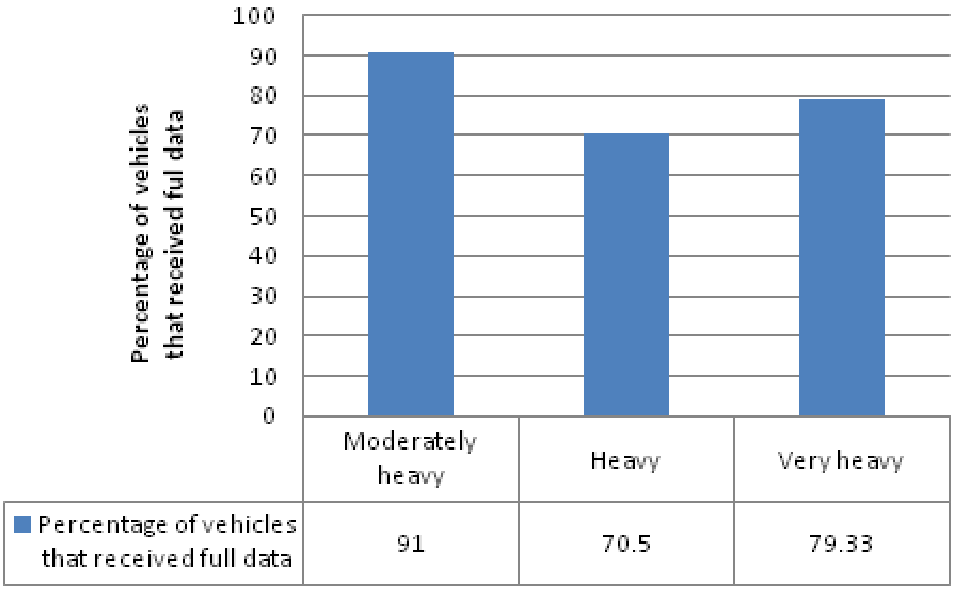


Figure 7.52: Percentage of vehicles that receive full amount of data

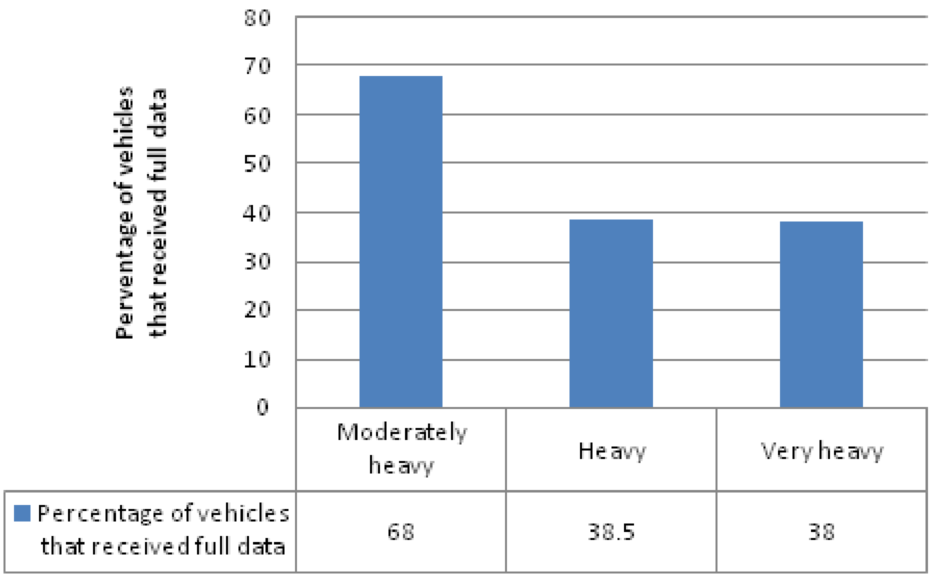


Figure 7.53: Average Data received by each vehicle

amount of data. There is an interesting anomaly here which should be noted. The percentage of data succesfully received first drops but then rises again slightly as traffic density rises from heavy to very heavy when it would perhaps be expected to drop further. This interesting effect can be explained - vehicles are trying to exchange information with an

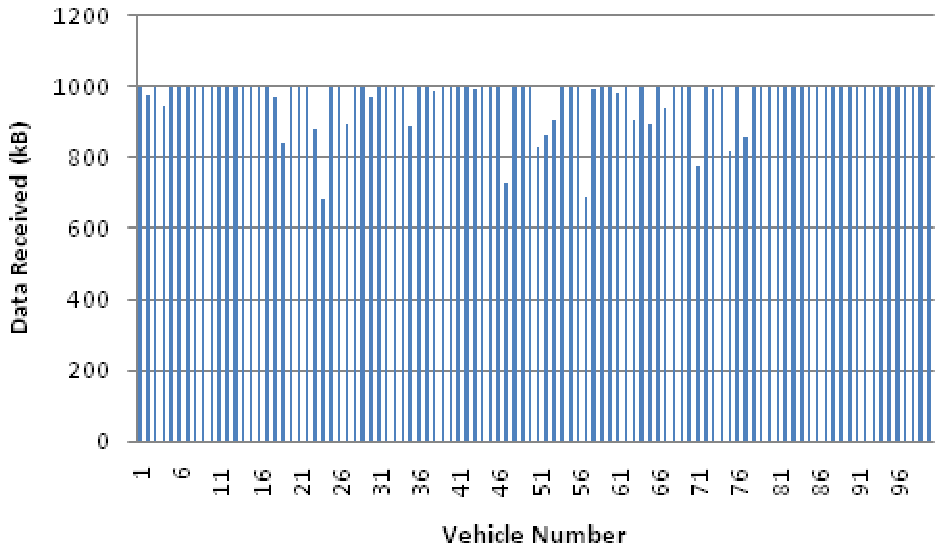


Figure 7.54: Data received by each vehicle when traffic is moderately heavy

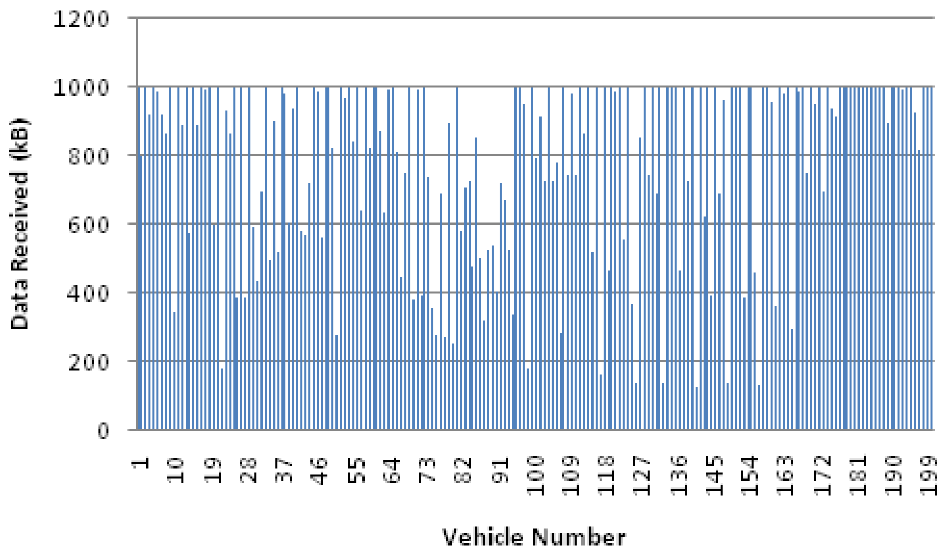


Figure 7.55: Data received by each vehicle when traffic is heavy

access point located at an intersection. When traffic is very heavy more vehicles are trying to exchange information however heavy traffic also slows the rate at which vehicles pass through the intersection. This means vehicles spend longer in range of the access point. In this instance the increased time spent in range compensates for the fact that there are

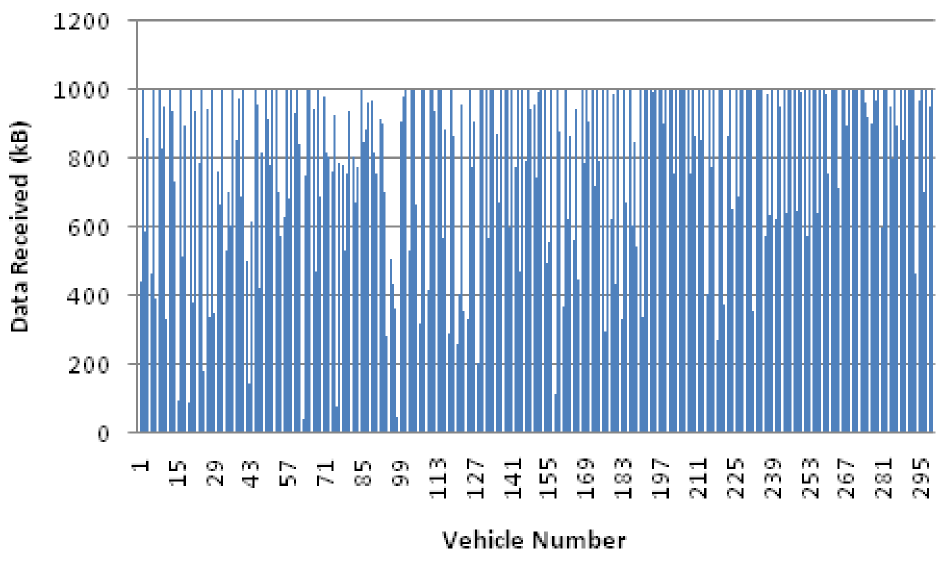


Figure 7.56: Data received by each vehicle when traffic is very heavy

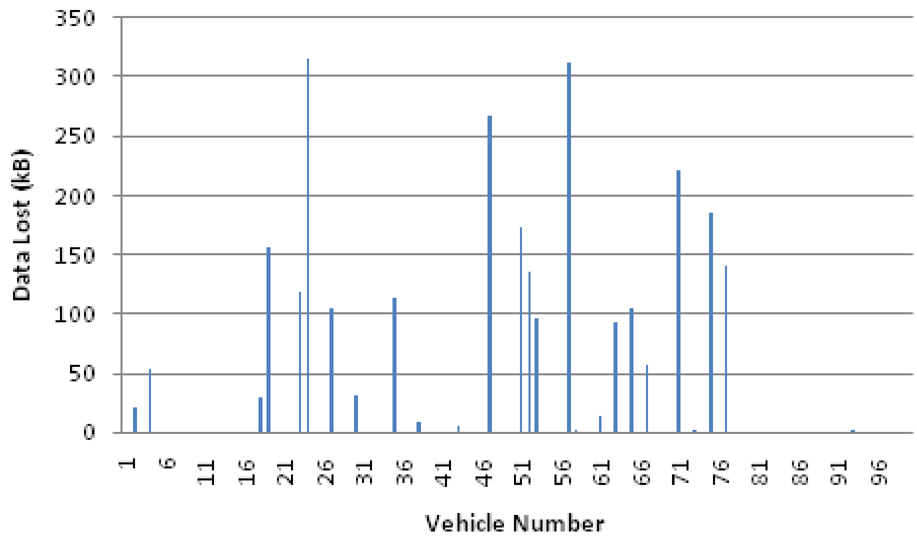


Figure 7.57: Data loss for each vehicle when traffic is moderately heavy

a greater number of vehicles contending resulting in an increase in data successfully received. This result is confirmed when you see that data loss increases when traffic density rises from heavy to very heavy in figures 7.58 and 7.59. This phenomenon was likely to occur if traffic slowed sufficiently as due to the size of vehicles and length of road available there is a maximum number of vehicles which may be in range at any one time while the

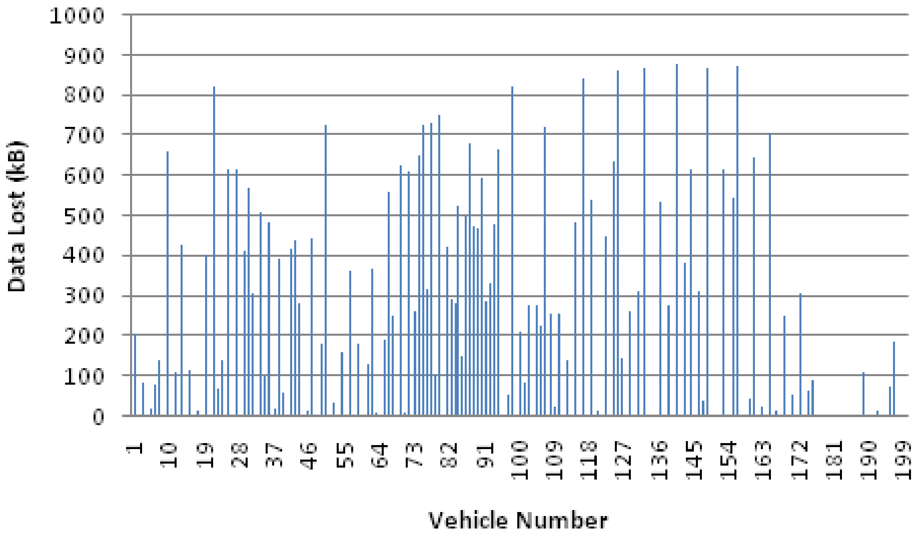


Figure 7.58: Data loss for each vehicle when traffic is heavy

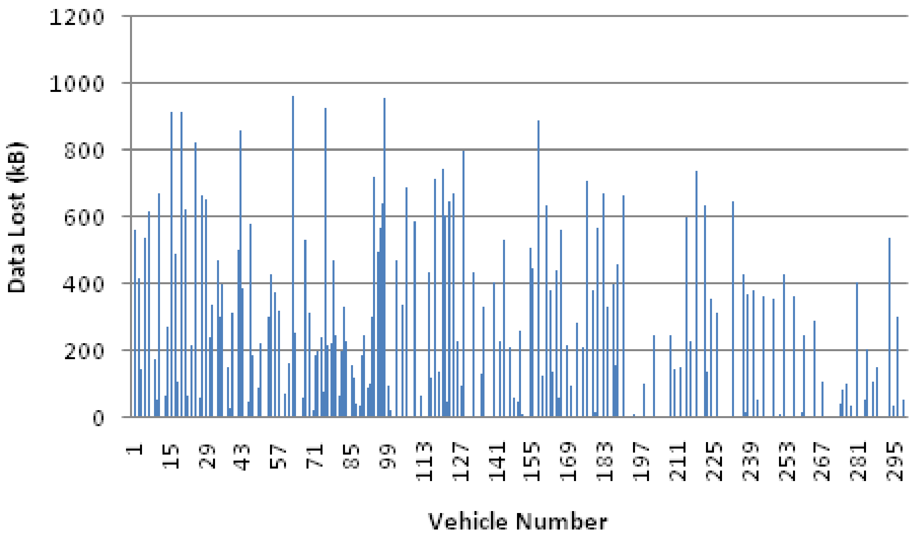


Figure 7.59: Data loss for each vehicle when traffic is very heavy

time spent waiting in range is not similarly constrained.

7.3.2.4 Analysis of Results

In order to analyse the results of the investigation into the levels of data exchange being achieved by the prioritised data exchange algorithm the data requirements of the connected

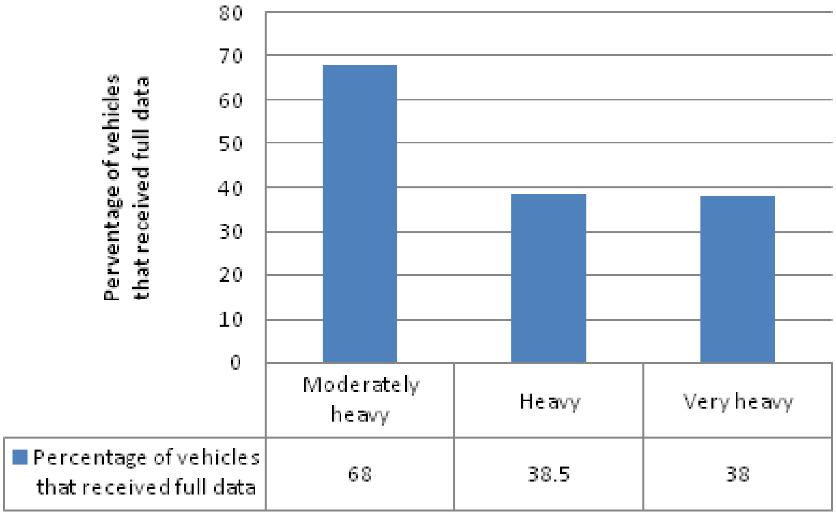


Figure 7.60: Percentage of vehicles that receive full amount of data

services being fed by it must be considered. The services being run can be categorised as traffic safety, traffic efficiency and value added services. The algorithm will send data to services in that order of priority. In the worst case scenario outlined above i.e. large numbers of cars present, all of which are trying to receive x amount of data from the roadside infrastructure you would want traffic safety services to be supported, as many traffic efficiency services as possible and any data left over for value added services would be a bonus.

To give an indication of the levels of data required for traffic safety and traffic efficiency services lets take the route management service powered by the best route selection algorithm described in the previous chapter as an example. The required data exchange between vehicles and roadside infrastructure, to provide the service was described in section 7.3.2 and is summarised in table 7.7 When a vehicle enters the range of a roadside unit it sends a service request. This service also requires vehicles to gather some data, the information vehicles are asked to collect is piggybacked onto the service request. In this case that consists of three parameters and the id for the road segment they describe. The data size is based on having a roadside unit at every intersection. It would increase if vehicles traversed more than one road segment between roadside units. The response from the roadside unit contains route information i.e. a list of the next 20 road segments the vehicle must take.

This allows a typical navigation unit to display a look ahead of the route being taken. The levels of data involved are relatively small for example 56 bytes to be sent and 320 bytes to be received. From this example we can see that intelligent transport services have relatively low data requirements.

	Data Size	**Note**
Data gathered and sent by vehicle	56 bytes per road segment described	If there is a communication hotspot at every intersection, some multiple of 56 bytes if density of hotspots is less
Data Received	16 bytes per road segment	for example if route is made up of 20 road segments then 320 bytes is sent, the amount of data sent drops as the vehicle approaches its destination

Table 7.7: Summary of data exchanged by vehicles for traffic management service

The results for case (1) where 250kB of data is transmitted to all vehicles show that on average with moderately heavy traffic, heavy traffic and very heavy traffic; 100%,99.2% and 99.7% of data is received respectively. In this instance a large number of intelligent transport services could be provided.

In cases (2) and (3) the amount of data transmitted to all vehicles is increased to 500kB and 1000kB. The amount of data getting through overall increases but lost data also goes up. This is where the prioritisation becomes important because as the data being lost has the lowest priority essential services can still be provided to vehicles and when the opportunity arises to receive more data value added services may also be supported.

7.3.3 Investigation into effect of data prioritisation

In this section we examine the implications of data prioritisation by examining a scenario involving a number of individual vehicles using data from the previous experiment.

In the scenario where traffic is very heavy and transmission of 1000kB is being attempted to each vehicle, vehicles 38,39 and 40 can receive 854, 976 and 686 kBs of data respectively. When these vehicles approach the roadside infrastructure, a safety message of 200kB is waiting to be sent by the roadside infrastructure with highest priority. Vehicles 38,39 and 40 run traffic efficiency services with data requirements of 400kB, 300kB and

450kB which are serviced next. That leaves vehicles with available data of 254kB, 476kb and 36kB respectively to attempt to meet the data requirements of any value-added services they may be running. The data figures are summarised in table 7.8.

Vehicle Number	Total Data (kB)	Safety Message (kB)	Traffic efficiency service (kB)	Remaining data for value added services (kB)
38	854	200	400	254
39	976	200	300	476
40	686	200	450	36

Table 7.8: Breakdown of total data available into prioritized components

What this example illustrates is that even though the data rate may fluctuate essential services such as traffic safety and traffic efficiency can be maintained and when the data rate is sufficient value added services may be supported.

7.3.4 Summary of Results

After analysis of results from testing of the prioritized data exchange algorithm the following has been shown. That even with worst case network scenario i.e. large numbers of cars present, all of which are trying to receive data from the roadside infrastructure, the algorithm can support a large number of intelligent transport type services. The importance of prioritisation in insuring that essential services are maintained despite fluctuations in data rate is demonstrated.

7.4 Chapter Summary

This chapter presented the experimental tests carried out in relation to the proposed Best Route Selection Algorithm and the Prioritised Data Exchange Algorithm. Detailed descriptions of the test setups were given. In-depth analysis of the experimental results was performed. The results which were outlined demonstrated the effectiveness of the proposed algorithms.

Chapter 8

Conclusion and Future Works

This chapter summarises the work in this thesis and presents possible future work.

8.1 Conclusion

This thesis describes a generic architecture for the provision of 'Smart City' services. These are services that are made possible by enabling elements of the urban environment for communication. An intelligent traffic management module is outlined for the provision of intelligent traffic management services within the Smart City architecture. A route management service, which is powered by a best route selection algorithm, is put forward as a prototypical traffic efficiency service for this module.

The best route selection algorithm aims to minimize journey times and traffic congestion as well as fuel consumption and emissions. These aims are mirrored by three cost components used by a fitness function in the algorithm; journey time cost, used capacity cost and fuel consumption and emissions cost. Results from simulation based testing show that the algorithm provides - shorter journey times, a reduction in fuel consumption and harmful emissions and also results in reduced fuel costs in comparison with two competing approaches. One of the competing approaches is a shortest path solution that uses estimated time values for road segments that are calculated by multiplying road segment length by speed limit and adding a time penalty based on the type of intersection found on the road segment. The second competing approach is a shortest path solution that uses the same "current" road segment time measurements as the proposed traffic management solution.

The best route selection algorithm has a number of settable parameters; k the number of shortest paths evaluated with the fitness function, w_1,w_2 and w_3 the weights associated with the three cost components of the fitness function and adaptivity, which may be switched on or off. Many different variants of the best route selection algorithm were tested to evaluate what settings led to the best performing variant of the TraffCon solution. For example the algorithm performs better with adaptation taking place than without. In the course of these tests it was shown that all three of the chosen costs have an associated benefit. The benefit associated with the fuel consumption and emissions cost was however not as significant as expected. This was attributed to elements of a real life road network which were missing from the simulation model. The fuel related cost component attempts to capture brake/accelerate behaviour of vehicles that impact fuel consumption and then reduce that type of behaviour. In real life elements such as pedestrians crossing the road, cyclists, parked cars and speed bumps contribute to this type of behaviour but they are not modelled in simulation.

The best route selection algorithm was also tested to see how it performed if the level of its penetration was reduced. The ability of the algorithm to reduce journey times in comparison with the competing approaches described above is shown to be robust to a reduction in the penetration level. For example the proposed solution operating at a penetration as low as 50% out performs the best competing approach operating at 100% penetration. The improvements in fuel economy however drops off quickly as the penetration is reduced. This is attributed to an inaccuracy in the measurements used to calculate used capacity cost when the penetration level drops. The used capacity cost was previously shown to be an important contributor to the way in which the algorithm improves fuel economy compared to competing approaches.

In order to provide the communications necessary to support the traffic management service proposed an architecture for a sparse roadside infrastructure-vehicle communications system is proposed. The IEEE 802.11p communications technology was used in the implementation for simulation based testing. IEEE 802.11p provides prioritisation of safety applications and a prioritised data exchange algorithm prioritises the communications necessary for any remaining traffic management or valued added services between vehicles and roadside infrastructure points. Test results show how it can support the provision a large number of intelligent transport type services while providing support for prioritisation.

The main contributions are summarised in the next section and potential future works in section 8.3.

8.2 Contributions

This thesis contributes to the state of the art in the area of wireless vehicular networks by introducing three novel aspects.

- The Smart City Service Oriented Architecture is a framework that enables the provision of a wide set of services to suitable mobile devices, supported by wireless communications. The basic functionality includes data harvesting, and data processing to gather information to a geospatial database. This information can be accessed by a wide range of services, which can be added in a modular fashion. Traffic management services are an example of typical services supported by this framework. The Smart City Service Oriented Architecture was detailed in 4.

- The Best Route Selection Algorithm powers a route management service for vehicles, which fits into the Smart City framework. It deploys in a sparse roadside infrastructure-based vehicle communications network. The novel aspect is that it minimizes journey time, traffic congestion as well as fuel consumption and emissions. The Best Route Selection Algorithm was outlined in chapter 6 and experimental test results and analysis of the performance of the algorithm was presented in 7.

- The Prioritized Data Exchange Algorithm supports the route management service and other intelligent transportation services that rely on wireless vehicular communication. It applies to sparse road sparse roadside infrastructure-based vehicle communications network. The novel aspect is that it enables increasing number of vehicles and growing amounts of data to be exchanged while prioritising data delivery based on service importance. The Prioritized Data Exchange Algorithm was outlined in chapter 6 and experimental test results and analysis of the performance of the algorithm was presented in 7.

8.3 Future Work

There are a number of logical progressions of the work outlined in this thesis. These range from - further studies of the existing solution, additional enhancements for the solution, to the application of the solution in a different setting. We will describe some of these potential future works in this section.

In section 7.2.6 it was stated that the fuel consumption and emissions cost parameter of the fitness function used in the best routes selection algorithm provided some benefit but was not as effective at improving fuel economy as expected. The cause of this is assumed to be that the cost component seeks to reduce brake/accelerate behaviours of vehicles that impact fuel consumption and is caused in real life by elements such as pedestrians crossing the road, cyclists, parked cars and speed bumps, which are not modelled in simulation. An informative future work would be to test this assumption. This may be done studying the effectiveness of the fuel consumption and emissions cost parameter of the fitness function using a simulation that models some of these additional interactions, which occur on real road networks. This would involve either modelling these elements in the existing simulation environment or implementing the solution in a different simulation environment that already models such elements of the urban environment. Alternatively the assumption could be tested by gathering data from real vehicles and investigating the significance of a vehicle's interaction with these elements of the urban environment.

In section 7.2.7 it was seen that the performance improvements achieved in fuel economy in comparison with competing approaches drops significantly when the penetration of the technology is reduced. This is attributed to an inaccuracy in the measurements used to calculate used capacity cost when the penetration level drops. The measurement of used capacity is compromised by the reduction in penetration level because if a vehicle is not reporting its whereabouts to the server, then its contribution to the used capacity cannot be measured. The used capacity cost was shown to be an important part of how the algorithm improves fuel economy compared to competing approaches in section 7.2.7. In order to make the solution more robust to a reduction in the level of its penetration an alternative means of counting vehicles that do support the technology, must be provided. A useful future work would be to explore ways in which this might be done. Possible approaches include using roadside sensors to count all vehicles or using vehicles employing the proposed solution to sense nearby vehicles not using it and count their contribution to used capacity.

Figure 8.1: Map of Dublin, Ireland showing areas marked as green zones

In this thesis we have looked at a city-wide system that runs a traffic management service that aims to reduce traffic congestion, minimize journey times for drivers and reduce fuel consumption across a given urban area. It would be interesting to extend the system to consider air quality or pollution as a global criterion. This could be done as an additional service or as an extension to the traffic management service, which attempts to minimize the contribution of vehicles to air pollution. How might this work? Certain areas of the city could be designated green zones as shown in figure 8.1. Only vehicles that meet certain emissions criteria are allowed enter these green zones for example zero emissions vehicles such as battery powered electric vehicles or vehicles that run on a hydrogen fuel cell. The central system has a record of the emission ratings of all vehicles so only vehicles which meet the criteria can be routed to or through the green-zones. Vehicles which do not meet the criteria are routed around the green zone to their destination or close to the green zone if their destination is in a green zone.

There are a number of interesting aspects to study in such a scenario. Considering that air pollutants can disperse through the atmosphere it would be interesting to see just how much air quality can be improved in the green-zones. It would also be interesting to vary what percentage of vehicles are rated zero emissions. Obviously if one hundred per cent of vehicles have zero emissions then the green zones have no affect and effectively don't exist. If however only a very small number of vehicles are rated zero emissions a study of the effect on traffic outside the green-zones will be of interest. Capacity will have been removed from the road network and you would expect increased traffic congestion in areas outside the green zones. It may also be the case that this causes air pollution outside the green-zones to increase.

The focus of this thesis is on the application of vehicular communication in urban areas. Road networks however spread far beyond the urban centres, they link urban centres and form vast networks, which criss cross countries and continents around the globe. In fact a substantial part of the road network is made up of those roads that are in the regions outside urban centres. In spite of this current works in the wireless vehicular networks space are notably urban-centric. A study into the potential impact of vehicular communication in the regions outside urban areas could prove useful. For example it could explore how public-private partnership transportation models, using vehicular communications services could improve accessibility of these remote regions. In the context of metropolitan regions future research could explore how public-private partnership transportation models using vehicular communications services could reduce urban sprawl. Combining these two ideas, the public-private partnership transportation models using vehicular communications services could prove useful to integrate the transport systems of cities and their outlying regions.

Appendix A: Calculating the k-shortest paths

The US Census Bureau's TIGER/Line files[1] provided the digital map base from which road maps used in this thesis were taken. In order to manage these road maps the Open-JUMP[2] geographical information system was used. OpenJUMP allowed the road maps to be viewed, edited, analysed etc. In order to calculate the k-shortest paths for every origin destination pair found in a given road map the OpenJUMP system was extended.The additional k-shortest paths functionality added to OpenJUMP can be seen in figure 8.2. To calculate the k-shortest paths the road map is selected in the right hand frame, the menu item for k-shortest paths is selected, the value of k is set and then the k-shortest paths can be calculated by clicking OK.

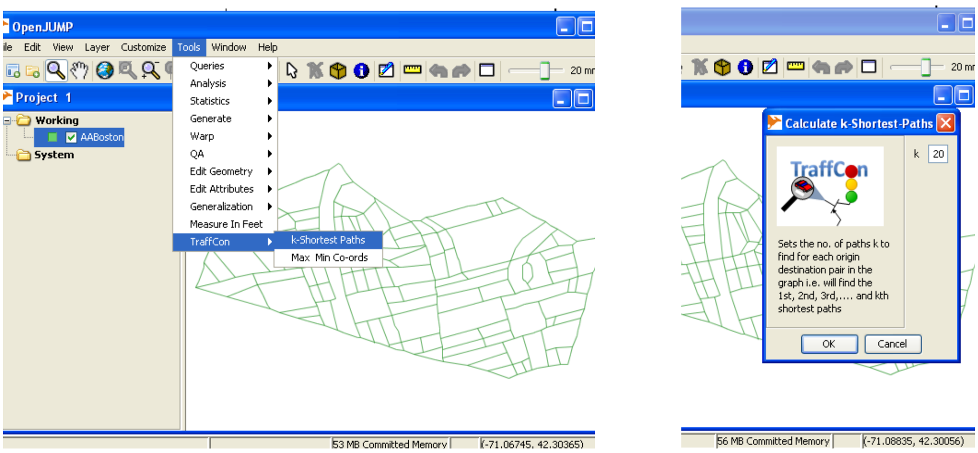


Figure 8.2: The k-shortest paths functionality in the OpenJUMP system

The documentation for developers found on the OpenJUMP website gives a detailed description of how to extend the system but a brief description will be given here to aid the understanding of how the k-shortest paths were calculated. OpenJUMP is written in Java and designed to be easily extensible. An OpenJUMP plug-in is an object that performs a single action, in response to a menu selection or a button press. The menu item or tool-bar button associated with the plug-in is added by an extension. An extension is a collection of

[1]http://www.census.gov/geo/www/tiger/
[2]http://www.openjump.org

classes and supporting resources that provides additional functionality to OpenJUMP. Extensions are packaged as JAR files. From the users perspective, extending OpenJUMP is as easy as copying an extension JAR file into OpenJUMP's plug-in directory. The OpenJUMP will search the JAR file for subclasses of the Extension class found in the OpenJUMP API (they must also be named Extension). It will then call a configure method on each Extension class it finds and add the new functionality to the system [166].

In order to add the new functionality a number of Java classes were written, the two main Java files are KShortestPlugIn.java which contains an implementation of the Floyd's shortest-path algorithm that generates a list of the k-shortest paths for the selected road map and TraffconExtension.java which is responsible for loading the new functionality. There are also a number of other helper classes for example graph algorithms such as Floyd's work on vertices and edges, consequently there is a class called Edge and a class called Vertex. The full list of helper class files is Edge.java, Vertex.java, Path.java, KPaths.java and Location.java. The source code for the extension which calculates the k-shortest paths follows, to fully understand the code it is first necessary to understand Floyd's algorithm a detailed description can be found in [54].

KShortestPlugIn.Java

```
//Plug-in which calculates k-shortest path for US Census Bureau Tiger/Line
//file road map. The map is first described as a graph and then the k-shortest
//paths are found using the generalised Floyd shortest-path algorithm

package traffcon;

import com.vividsolutions.jump.feature.Feature;
import com.vividsolutions.jump.feature.FeatureCollectionWrapper;
import com.vividsolutions.jump.task.TaskMonitor;
import com.vividsolutions.jump.workbench.model.Layer;
import com.vividsolutions.jump.workbench.plugin.PlugInContext;
import com.vividsolutions.jump.workbench.plugin.ThreadedBasePlugIn;
import com.vividsolutions.jump.workbench.ui.GUIUtil;
import com.vividsolutions.jump.workbench.ui.MultiInputDialog;
```

```
import com.vividsolutions.jump.workbench.ui.plugin.FeatureInstaller;
import java.io.*;
import java.util.*;
import javax.swing.ImageIcon;
import javax.swing.JFileChooser;

public class KShortestPlugIn extends ThreadedBasePlugIn {

public KShortestPlugIn() {
}

public void initialize(PlugInContext context) throws Exception {
context.getFeatureInstaller().addMainMenuItem(this,
new String[] { "Tools", "TraffCon" }, "k-Shortest Paths",
false, null, null);
}

public boolean execute(PlugInContext context) throws Exception {
MultiInputDialog dialog = new MultiInputDialog(context
.getWorkbenchFrame(), "Calculate k-Shortest-Paths", true);
setDialogValues(dialog, context);
GUIUtil.centreOnWindow(dialog);
dialog.setVisible(true);

if (!dialog.wasOKPressed()) {
return false;
} else {
getDialogValues(dialog);
return true;
}
}

public void run(TaskMonitor monitor, PlugInContext context)
throws Exception {
FeatureCollectionWrapper FCWrap = context.getSelectedLayers()[0]
.getFeatureCollectionWrapper();
```

```
A = null;
B = null;
edge = null;
vertexList = new Vector();
edgeList = new Vector();
adjacencyList = new Hashtable();
incidenceList = new Hashtable();
pathList = new Hashtable();

//TODO File Location hard coded add file chooser dialog for better usability
//and to avoid File not Found errors when installed on other systems
File shapefile = new File(
"C:\\Program Files\\OpenJUMP\\shapefile.dat");
outputStream = new PrintWriter(new FileWriter(shapefile));

buildGraph(FCWrap, outputStream);

for (Iterator i = edgeList.iterator(); i.hasNext(); outputStream
.println(i.next().toString()));

calculateKShortestPaths();
buildHashtable(vertexList.size());

if (outputStream != null)
outputStream.close();

return;
}

public void buildGraph(FeatureCollectionWrapper FCWrap,
PrintWriter outputStream) {
Iterator i = FCWrap.iterator();
int id = 0;
Feature f;
for (; i.hasNext(); buildIncidenceList(outputStream, f, id)) {
id++;
```

```
f = (Feature) i.next();
buildAdjacencyList(outputStream, f);
}
}

public void buildAdjacencyList(PrintWriter outputStream, Feature f) {
StringTokenizer tok = new StringTokenizer(f.getString(0), " (),");
for (int j = 0; tok.hasMoreTokens(); j++)
if (j == 0) {
tok.nextToken();
double a = (new Double(tok.nextToken())).doubleValue();
double b = (new Double(tok.nextToken())).doubleValue();
A = new Vertex(new Location(a, b));
} else {
double a = (new Double(tok.nextToken())).doubleValue();
double b = (new Double(tok.nextToken())).doubleValue();
B = new Vertex(new Location(a, b));
}

if (adjacencyList.containsKey(A.getID())) {
if (!((Vector) adjacencyList.get(A.getID())).contains(B))
((Vector) adjacencyList.get(A.getID())).add(B);
} else {
Vector adjacentVertices = new Vector();
adjacentVertices.add(B);
adjacencyList.put(A.getID(), adjacentVertices);
vertexList.add(A);
outputStream.println((new StringBuilder("Vertex A:")).append(
A.toString()).toString());
}
if (adjacencyList.containsKey(B.getID())) {
if (!((Vector) adjacencyList.get(B.getID())).contains(A))
((Vector) adjacencyList.get(B.getID())).add(A);
} else {
Vector adjacentVertices = new Vector();
adjacentVertices.add(A);
```

```
adjacencyList.put(B.getID(), adjacentVertices);
vertexList.add(B);
outputStream.println((new StringBuilder("Vertex B:")).append(
B.toString()).toString());
}
}

public void buildIncidenceList(PrintWriter outputStream, Feature f, int id) {
outputStream.println(((Double) f.getAttribute("LENGTH")).doubleValue());
edge = new Edge(id, A, B, ((Double) f.getAttribute("LENGTH"))
.doubleValue());
edgeList.add(edge);
if (incidenceList.containsKey(A.getID())) {
((Vector) incidenceList.get(A.getID())).add(edge);
} else {
Vector incidentEdges = new Vector();
incidentEdges.add(edge);
incidenceList.put(A.getID(), incidentEdges);
}
if (incidenceList.containsKey(B.getID())) {
((Vector) incidenceList.get(B.getID())).add(edge);
} else {
Vector incidentEdges = new Vector();
incidentEdges.add(edge);
incidenceList.put(B.getID(), incidentEdges);
}
}

private void calculateKShortestPaths() {
int M = vertexList.size();
kPaths = new KPaths[M][M];
for (int i = 0; i < M; i++) {
for (int j = 0; j < M; j++)
kPaths[i][j] = new KPaths(noOfPaths);
}
```

```
for (int i = 0; i < M; i++) {
for (int j = 0; j < M; j++)
if (i == j)
kPaths[i][j].zeroes();
else if (((Vector) adjacencyList.get(((Vertex) vertexList
.elementAt(i)).getID())).contains(vertexList
.elementAt(j))) {
for (Iterator it = ((Vector) incidenceList
.get(((Vertex) vertexList.elementAt(i)).getID()))
.iterator(); it.hasNext();) {
Edge e = (Edge) it.next();
if ((e.getA().equals(vertexList.elementAt(j)) || e
.getB().equals(vertexList.elementAt(j)))
&& e.getWeight() < kPaths[i][j].getPaths()[noOfPaths - 1]
.getWeight())
kPaths[i][j].addPath(new Path(e.getWeight(),
(new Integer(e.getID())).toString()),
outputStream);
}
}
}

kPathsPlusOne = kPaths;
for (int k = 0; k < M; k++) {
for (int i = 0; i < M; i++) {
for (int j = 0; j < M; j++) {
for (int l = 0; l < noOfPaths; l++) {
for (int m = 0; m < noOfPaths; m++)
if (kPaths[i][k].getPaths()[l].getWeight()
+ kPaths[k][j].getPaths()[m].getWeight() < kPaths[i][j]
.getPaths()[noOfPaths - 1].getWeight())
kPathsPlusOne[i][j]
.addPath(
new Path(
kPaths[i][k].getPaths()[l]
.getWeight()
```

```
+ kPaths[k][j]
.getPaths()[m]
.getWeight(),
(new StringBuilder(
String
.valueOf(kPaths[i][k]
.getPaths()[l]
.getEdges())))
.append(",")
.append(
kPaths[k][j]
.getPaths()[m]
.getEdges())
.toString()),
outputStream);

}
}
}

kPaths = kPathsPlusOne;
}

printShortestPaths(M);
}

private void buildHashtable(int M) {
for (int i = 0; i < M; i++) {
for (int j = 0; j < M; j++)
pathList.put((new StringBuilder("(")).append(i).append(",")
.append(j).append(")").toString(), kPaths[i][j]
.getPaths());
}

try {
File paths = new File(
```

```
"C:\\Documents and Settings\\KC\\Desktop\\Dev\\GIS\\paths.dat");
FileOutputStream fileOut = new FileOutputStream(paths);
ObjectOutputStream out = new ObjectOutputStream(fileOut);
System.out.println("Writing Hashtable Object...");
out.writeObject(pathList);
System.out.println("Closing all output streams...\n");
out.close();
fileOut.close();
} catch (FileNotFoundException e) {
e.printStackTrace();
} catch (IOException e) {
e.printStackTrace();
}
}

private void setDialogValues(MultiInputDialog dialog, PlugInContext context) {
dialog.setSideBarImage(new ImageIcon(getClass().getResource(
"Traffcon_Logo1.png")));
dialog
.setSideBarDescription("Sets the no. of paths k to find for each origin destination pair
dialog
.addIntegerField("k", 1, 2,
"Sets the no. of paths k to find for each origin destination pair in the graph");
}

private void getDialogValues(MultiInputDialog dialog) {
noOfPaths = dialog.getInteger("k");
}

public void printShortestPaths(int M) {
for (int i = 0; i < M; i++) {
for (int j = 0; j < M; j++)
outputStream.println((new StringBuilder("(")).append(i).append(
",").append(j).append(")  ").append(
kPaths[i][j].printWeights()).toString());
```

```
}

for (int i = 0; i < M; i++) {
for (int j = 0; j < M; j++)
outputStream.println((new StringBuilder("(")).append(i).append(
",").append(j).append(")  ").append(
kPaths[i][j].printPaths()).toString());
}
}

private int noOfPaths;
private Vertex A;
private Vertex B;
private Edge edge;
private Vector vertexList;
private Vector edgeList;
private Hashtable adjacencyList;
private Hashtable incidenceList;
private Hashtable pathList;
private KPaths kPaths[][];
private KPaths kPathsPlusOne[][];
private PrintWriter outputStream;
}
```

TraffConExtension.Java

```
package traffcon;

import com.vividsolutions.jump.workbench.plugin.Extension;
import com.vividsolutions.jump.workbench.plugin.PlugInContext;

public class TraffconExtension extends Extension {

public TraffconExtension() { }

public void configure(PlugInContext context) throws Exception {
```

```
(new KShortestPlugIn()).initialize(context);
(new MaxMinPlugIn()).initialize(context);
}
}
```

Edge.Java

```
//Java representation of edge for creating graph

package traffcon;

public class Edge {

Edge(int ID, Vertex A, Vertex B, double weight) {
this.ID = ID;
this.A = A;
this.B = B;
this.weight = weight;
}

public Vertex getA() { return A;
}

public void setA(Vertex a) { A = a;
}

public Vertex getB() { return B;
}

public void setB(Vertex b) { B = b;
}

public int getID() { return ID;
}

public void setID(int id) { ID = id;
```

```
}

public double getWeight() { return weight;
}

public void setWeight(double weight) {this.weight = weight;
}

public String toString() {
return (new StringBuilder(String.valueOf(A.toString()))).append(
B.toString()).toString();
}

private int ID;
private Vertex A;
private Vertex B;
private double weight;
}
```

Vertex.Java

```
//Java representation of vertex for creating graph

package traffcon;

import java.util.Set;

public class Vertex {

public Vertex(Location loc) {
this.loc = loc;
if (loc.getLongitude() > 0.0D && loc.getLatitude() > 0.0D)
ID = (new StringBuilder("+")).append(
Rounding.round(loc.getLongitude(), 8)).append("+").append(
Rounding.round(loc.getLatitude(), 8)).toString();
else if (loc.getLongitude() < 0.0D && loc.getLatitude() > 0.0D)
```

```
ID = (new StringBuilder(String.valueOf(Rounding.round(loc
.getLongitude(), 8)))).append("+").append(
Rounding.round(loc.getLatitude(), 8)).toString();
else if (loc.getLongitude() > 0.0D && loc.getLatitude() < 0.0D)
ID = (new StringBuilder("+")).append(
Rounding.round(loc.getLongitude(), 8)).append(
Rounding.round(loc.getLatitude(), 8)).toString();
else
ID = (new StringBuilder(String.valueOf(new String((new Double(
Rounding.round(loc.getLongitude(), 8))).toString()))))
.append(
new String((new Double(Rounding.round(loc
.getLatitude(), 8))).toString()))
.toString();
}

public Vertex(double longitude, double lattitude) {
loc = new Location(longitude, lattitude);
if (loc.getLongitude() > 0.0D && loc.getLatitude() > 0.0D)
ID = (new StringBuilder("+")).append(
Rounding.round(loc.getLongitude(), 8)).append("+").append(
Rounding.round(loc.getLatitude(), 8)).toString();
else if (loc.getLongitude() < 0.0D && loc.getLatitude() > 0.0D)
ID = (new StringBuilder(String.valueOf(Rounding.round(loc
.getLongitude(), 8)))).append("+").append(
Rounding.round(loc.getLatitude(), 8)).toString();
else if (loc.getLongitude() > 0.0D && loc.getLatitude() < 0.0D)
ID = (new StringBuilder("+")).append(
Rounding.round(loc.getLongitude(), 8)).append(
Rounding.round(loc.getLatitude(), 8)).toString();
else
ID = (new StringBuilder(String.valueOf(new String((new Double(
Rounding.round(loc.getLongitude(), 8))).toString()))))
.append(
new String((new Double(Rounding.round(loc
.getLatitude(), 8))).toString()))
```

```
.toString();
}

public Set getAdjacentVertices() { return adjacentVertices;
}

public void setAdjacentVertices(Set adjacentVertices) {
this.adjacentVertices = adjacentVertices;
}

public void addAdjacentVertex(Vertex adjacentVertex) {
adjacentVertices.add(adjacentVertex);
}

public String getID() { return ID;
}

public void setID(String id) { ID = id;
}

public Location getLoc() { return loc;
}

public void setLoc(Location loc) {  this.loc = loc;
}

public String toString() { return ID;
}

public boolean equals(Object o) {
return ((Vertex) o).getLoc().equals(loc);
}

private String ID;
private Location loc;
private Set adjacentVertices;
```

```
}
```

Path.Java

```
// Class for description of a Path from Origin A to Destination B

package traffcon;

import java.io.Serializable;

public class Path implements Comparable, Serializable {

Path(double weight, String edges) {
this.weight = weight;
this.edges = edges;
}

public String getEdges() { return edges;
}

public void setEdges(String edges) { this.edges = edges;
}

public double getWeight() { return weight;
}

public void setWeight(double weight) { this.weight = weight;
}

public int compareTo(Path p) {
int lastCmp = (new Double(weight)).compareTo(Double.valueOf(p
.getWeight()));
return lastCmp == 0 ? edges.compareTo(p.getEdges()) : lastCmp;
}

public int compareTo(Object obj) {
```

```
return compareTo((Path) obj);
}

static final long serialVersionUID = 1L;
private double weight;
private String edges;
}
```

KPaths.Java

```
//The k-shortest paths for all origin destination pairs in a graph

package traffcon;

import java.io.PrintWriter;
import java.util.Arrays;
import java.util.Collections;
import java.util.StringTokenizer;


public class KPaths {

KPaths(int k) {
noOfPaths = k;
paths = new Path[k];
for (int i = 0; i < noOfPaths; i++)
paths[i] = new Path((1.0D / 0.0D), "");
}

public void addPath(Path p, PrintWriter outputStream) {
boolean duplicate = false;
boolean hasloop = false;
for (int i = 0; i < noOfPaths - 1; i++)
if (p.getWeight() == paths[i].getWeight())
duplicate = true;
```

```
StringTokenizer tok = new StringTokenizer(p.getEdges(), ",");
int current;
for (int last = (new Integer(tok.nextToken())).intValue(); tok
.hasMoreTokens(); last = current) {
current = (new Integer(tok.nextToken())).intValue();
if (current != last)
continue;
hasloop = true;
break;
}

if (!duplicate && !hasloop) {
paths[noOfPaths - 1] = p;
java.util.List pathList = Arrays.asList(paths);
Collections.sort(pathList);
}
}

public void zeroes() {
for (int i = 0; i < noOfPaths; i++)
paths[i] = new Path(0.0D, "");

}

public int getNoOfPaths() { return noOfPaths;
}

public void setNoOfPaths(int noOfPaths) {
this.noOfPaths = noOfPaths;
}

public Path[] getPaths() { return paths;
}

public void setPaths(Path paths[]) { this.paths = paths;
}
```

```
public String printWeights() {
String weights = "";
for (int i = 0; i < noOfPaths; i++)
weights = (new StringBuilder(String.valueOf(weights))).append("[")
.append(paths[i].getWeight()).append("]").toString();

return weights;
}

public String printPaths() {
String pathlist = "";
for (int i = 0; i < noOfPaths; i++)
pathlist = (new StringBuilder(String.valueOf(pathlist)))
.append("[").append(paths[i].getEdges()).append("]")
.toString();

return pathlist;
}

private int noOfPaths;
private Path paths[];
}
```

Location.Java

```
//longitude-latitude location

package traffcon;

public class Location {

Location(double x, double y) {
longitude = x;
latitude = y;
}
```

```
public double getLatitude() {
return latitude;
}

public void setLatitude(double latitude) {
this.latitude = latitude;
}

public double getLongitude() {
return longitude;
}

public void setLongitude(double longitude) {
this.longitude = longitude;
}

public boolean equals(Object o) {
return ((Location) o).getLatitude() == latitude
&& ((Location) o).getLongitude() == longitude;
}

private double longitude;
private double latitude;
}
```

Appendix B: SWANS++ settings for simulation

Modifying SWANS

The JistExperiment class is used to configure experiments so if a modification to SWANS requires a new configurable parameter then it should be added to the JistExperiment class. This parameter can then be modified using the XML configuration file for example;

```
<!--New parameters to configure TraffCon Algorithm-->
<void property="k">
   <int>10</int>
</void>

<void property="w1">
   <float>.5</float>
</void>

<void property="w2">
   <float>0</float>
</void>

<void property="w3">
   <float>0</float> </void>
```

One approach to simulating a traffic management solution in SWANS is to modify the RoadSegment class which is used to describe individual road segments during simulation so it can store data about the state of the road network collected by vehicles. The traffic management algorithm can then be implemented as a new mobility model which is a modified version of the STRAW mobility model (StreetMobilityOD class). Vehicles which use the traffic management algorithm follow the same vehicular mobility rules as other vehicles for interactions with other vehicles, at traffic lights, stop signs etc. but the route taken from origin to destination is chosen according to the traffic management algorithm.

Sample XML configuration file

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<java version="1.4.2_05" class="java.beans.XMLDecoder">
 <object class="driver.JistExperiment">

 <!-- timing -->
  <void property="duration">
   <int>900</int>
  </void>
  <void property="resolutionTime">
   <int>60</int>
  </void>
  <void property="startTime">
   <int>60</int>
  </void>

 <!-- environment -->
  <void property="exponent">
   <double>2.8</double>
  </void>
  <void property="stdDeviation">
   <double>6.0</double>
  </void>
    <void property="loss">
   <int>0</int>
  </void>
  <void property="pathloss">
   <int>1</int>
  </void>
  <void property="transmit">
   <double>16</double>
  </void>

   <!-- mobility options -->
  <void property="mobility">
   <int>6</int>
  </void>
```

```
<!--street specific ops-->
<void property="degree">
 <int>4</int>
</void>
<void property="probability">
 <double>0.3</double>
</void>
  <void property="minLat">
 <float>42.36134</float>
</void>
<void property="minLong">
 <float>-71.05870</float>
</void>
    <void property="maxLat">
 <float>42.36808</float>
</void>
<void property="maxLong">
 <float>-71.05024</float>
</void>
<void property="segmentFile">
 <string>suffolk/segments.dat</string>
</void>
  <void property="streetFile">
 <string>suffolk/names.dat</string>
</void>
  <void property="shapeFile">
 <string>suffolk/chains.dat</string>
</void>

 <!-- node options -->
 <void property="nodes">
 <int>50</int>
</void>
 <void property="transmitters">
 <int>10</int>
</void>
```

```
   <void property="sendRate">
  <double>1.0</double>
 </void>
  <!-- placement options -->
  <void property="placement">
  <int>3</int>
 </void>
   <void property="spatial_mode">
  <int>2</int>
 </void>

 <!-- other options -->
 <void property="port">
  <int>3001</int>
 </void>
 <void property="protocol">
  <int>123</int>
 </void>
   <void property="help">
  <boolean>false</boolean>
 </void>
</object> </java>
```

Appendix C: IEEE 802.11p ns-2 settings for Simulation

Mobility Settings

The mobility pattern of a wireless node is another important setting which must be considered. In ns-2 a node is given an initial location within a topography. From here the node can be moved using a series of variations on the following type of command -

```
$ns_ at  t  "$node_(1)  setdest  x  y  v"
```

This command causes node one to begin moving in the direction of co-ordinate (x, y) at time t with velocity v m/s. In the simple example shown hereafter two nodes are placed in a 500m x 500m field. One node is stationary in the centre and the other node moves towards the node from the bottom left corner at the walking type speed of 1.4722 m/s (5.3kph). The node then moves back to its start position at the same speed.

```
# Provide initial (X,Y, for now Z=0) co-ordinates for nodes
$node_(0) set X_ 250.0
$node_(0) set Y_ 250.0
$node_(0) set Z_ 0.0

$node_(1) set X_ 0.0
$node_(1) set Y_ 0.0
$node_(1) set Z_ 0.0

# Now produce some simple node movements
# In this exmple node 0 is stationary
# Node_(1) starts to move towards node_(0)

$ns_ at 0.0 "$node_(1) setdest 250.0 250.0 1.4722"

# Node_(1) then starts to move away from node_(0)

$ns_ at 240.0 "$node_(1) setdest 0.0 0.0 1.4722"
```

In order to simulate vehicular mobility in a large urban area, a larger field size is used and a very large number of these type of commands is required having considerably more complex variation in velocity.

Sample TCL file for 802.11p settings in ns-2.34

```
# 802.11p default parameters

# MAC Settings

Mac/802_11Ext set CWMin_                    15
Mac/802_11Ext set CWMax_                    1023
Mac/802_11Ext set SlotTime_                 0.000013
Mac/802_11Ext set SIFS_                     0.000032
Mac/802_11Ext set ShortRetryLimit_          7
Mac/802_11Ext set LongRetryLimit_           4
Mac/802_11Ext set HeaderDuration_           0.000040
Mac/802_11Ext set SymbolDuration_           0.000008
Mac/802_11Ext set BasicModulationScheme_    0
Mac/802_11Ext set use_802_11a_flag_         true
Mac/802_11Ext set RTSThreshold_             2346
Mac/802_11Ext set MAC_DBG                   0

# PHY Settings

Phy/WirelessPhyExt set CSThresh_            3.9810717055349694e-13
Phy/WirelessPhyExt set Pt_                  0.1
Phy/WirelessPhyExt set freq_                5.9e+9
Phy/WirelessPhyExt set noise_floor_         1.26e-13
Phy/WirelessPhyExt set L_                   1.0
Phy/WirelessPhyExt set PowerMonitorThresh_  3.981071705534985e-18
Phy/WirelessPhyExt set HeaderDuration_      0.000040
```

```
Phy/WirelessPhyExt set BasicModulationScheme_   0
Phy/WirelessPhyExt set PreambleCaptureSwitch_   1
Phy/WirelessPhyExt set DataCaptureSwitch_       1
Phy/WirelessPhyExt set SINR_PreambleCapture_    3.1623;
Phy/WirelessPhyExt set SINR_DataCapture_        10.0;
Phy/WirelessPhyExt set trace_dist_              1e6
Phy/WirelessPhyExt set PHY_DBG_                 0


# RF Propagation Settings

Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0

Propagation/Nakagami set use_nakagami_dist_ false
Propagation/Nakagami set gamma0_ 2.0
Propagation/Nakagami set gamma1_ 2.0
Propagation/Nakagami set gamma2_ 2.0

Propagation/Nakagami set d0_gamma_ 200
Propagation/Nakagami set d1_gamma_ 500

Propagation/Nakagami set m0_  1.0
Propagation/Nakagami set m1_  1.0
Propagation/Nakagami set m2_  1.0

Propagation/Nakagami set d0_m_ 80
Propagation/Nakagami set d1_m_ 200
```

Bibliography

[1] Cooperative Vehicle-Infrastructure Systems. http://www.cvisproject.org.

[2] Iso tc 204: Intelligent transport systems, 1992.

[3] ABI Research. Connected navigation devices. Technical report, ABI Research, April 2008.

[4] R. Akcelik, M. Besley, and E. Chung. An evaluation of scats master isolated control. In *Proc 19th ARRB Transport Research Conf.*, Sydney, Australia, 1998.

[5] R. Akcilek and M. Besley. Operating cost, fuel consumption and emission models in aaSIDRA and aaMotion. In *25th Conference of Australian Institutes of Transport Research*, Adelaide, Australia, 2003.

[6] I.F. Akyildiz and Xudong Wang. A survey on wireless mesh networks. *Communications Magazine, IEEE*, 43(9):S23–S30, Sept. 2005.

[7] AMPO. Noteworthy IMO practices in transportation-land use planning integration. Technical report, Association of Metropolitan Planning Organizations, April 2004.

[8] Mitsuo Arino. Its policy in japan and smartway. In *15th World Congress on ITS*. Japanese Ministry of Land, Infrastructure and Transport, November 2008.

[9] ASTM. Standard specification for telecomunications and information exchange between roadside and vehicle systems - 5ghz band dedicated short range communications (dsrc) medium access control (mac) and physical layer (phy). *ASTM E2213-03*, September 2003.

[10] M. Audeh. Metropolitan-scale wi-fi mesh networks. *Computer*, 37(12):119–121, Dec. 2004.

[11] Andrew Bacha, Cheryl Bauman, Ruel Faruque, Michael Fleming, Chris Terwelp, Charles F. Reinholtz, Dennis Hong, Al Wicks, Thomas Alberi, David Anderson, Stephen Cacciola, Patrick Currier, Aaron Dalton, Jesse Farmer, Jesse Hurdus, Shawn Kimmel, Peter King, Andrew Taylor, David Van Covern, and Mike Webster. Odin: Team victortango's entry in the darpa urban challenge. *J. Field Robotics*, 25(8):467–492, 2008.

[12] Aruna Balasubramanian, Brian Neil Levine, and Arun Venkataramani. DTN Routing as a Resource Allocation Problem. *IEEE/ACM Transactions on Networking*, To Appear 2009.

[13] Nilanjan Banerjee, Mark D. Corner, and Brian Neil Levine. An Energy-Efficient Architecture for DTN Throwboxes. *IEEE/ACM Transactions on Networking*, page 14 pages, To Appear 2009.

[14] Rimon Barr, Zygmunt J. Haas, and Robbert van Renesse. *Handbook on Theoretical and Algorithmic Aspects of Sensor, Ad hoc Wireless, and Peer-to-Peer Networks.*, chapter 19: Scalable Wireless Ad hoc Network Simulation, pages 297–311. CRC Press, 2005.

[15] J. Blau. Car talk - [update]. *Spectrum, IEEE*, 45(10):16–16, Oct. 2008.

[16] P.T. Blythe. Combatting congestion: the role of road traffic informatics. In *Traffic Congestion - Engineering the Solution, IEE Colloquium on*, pages 3/1–314, 15 May 1991.

[17] Jonathan Bohren, Tully Foote, Jim Keller, Alex Kushleyev, Daniel D. Lee, Alex Stewart, Paul Vernaza, Jason C. Derenick, John R. Spletzer, and Brian Satterfield. Little ben: The ben franklin racing team's entry in the 2007 darpa urban challenge. *J. Field Robotics*, 25(9):598–614, 2008.

[18] Richard Bossom, Roberto Brignolo, Thierry Ernst, Knut Evensen, Alexander Frtscher, Wolfgang Hfs, Juhani Jskelinen, Zeljko Jeftic, Paul Kompfner, Timo Kosch, Ilse Kulp, Antonio Kung, Abdel-Kader Mokaddem, Andreas Schalk, Elisabeth Uhlemann, and Christian Wewetzer. European its communication architecture overall framework proof of concept implementation. Technical report, COMeSafety, March 2009.

[19] Leo Breiman. The poisson tendency in traffic distribution. *The Annals of Mathematical Statistics*, 34(1):308–311, 1963.

[20] David Bretherton, Keith Wood, and Neil Raha. Traffic monitoring and congestion management in the scoot urban traffic control system. *Traffic Research Record*, 1634:118–122, 1998.

[21] R. Bruno, M. Conti, and E. Gregori. Mesh networks: commodity multihop ad hoc networks. *Communications Magazine, IEEE*, 43(3):123–131, March 2005.

[22] Bill Bryson. *A Short History of Nearly Everything*. Black Swan, 2004.

[23] J Burgess, B Gallagher, D Jensen, and B.N. Levine. Maxprop: Routing for vehicle-based disruption-tolerant networks. In *25th IEEE International Conference on Computer Communications*, Barcelona, Spain, April 2006.

[24] Alasdair Cain. Achieving majority public support for urban road pricing: Preserving the driver's right to choose. *Transportation Research Record*, 1932:119–128, 2005.

[25] Murat Caliskan, Andreas Barthels, Björn Scheuermann, and Martin Mauve. Predicting parking lot occupancy in vehicular ad hoc networks. In *VTC Spring*, pages 277–281, 2007.

[26] Murat Caliskan, Daniel Graupner, and Martin Mauve. Decentralized discovery of free parking places. In *VANET'06*, pages 30–39, Los Angeles, CA, USA, Sept. 2006.

[27] UK Cambridgeshire County Council. The 2006 traffic monitoring report. Technical report, Cambridgeshire County Council, 2006.

[28] Xian-Bin Cao, Hong Qiao, and J. Keane. A low-cost pedestrian-detection system with a single optical camera. *Intelligent Transportation Systems, IEEE Transactions on*, 9(1):58–67, March 2008.

[29] Ennio Cascetta. *Transportation Systems Engineering: Theory and Methods*. Springer, 2001.

[30] Braudy Cash. WAVE background information. In *WAVE Study Group*, 2004.

[31] A. Chen, B. Khorashadi, Chen-Nee Chuah, D. Ghosal, and M. Zhang. Smoothing vehicular traffic flow using vehicular-based ad hoc networking &computing grid (vgrid). In *Intelligent Transportation Systems, 2006. Proceedings. 2006 IEEE*, pages 349–354, 2006.

[32] Qi Chen, Felix Schmidt-Eisenlohr, Daniel Jiang, Marc Torrent-Moreno, Luca Delgrossi, and Hannes Hartenstein. Overhaul of IEEE 802.11 modeling and simulation in ns-2. In *MSWiM '07: Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*, pages 159–168, New York, NY, USA, 2007. ACM.

[33] David R. Choffnes and Fabin E. Bustamante. STRAW - an integrated mobility and traffic model for VANETs. In *International Command and Control Research and Technology Symposium*, McLean, VA, USA, June 2005.

[34] C-Nee Chuah, H. Du, D. Ghosal, B. Khorashadi, B. Liu, C. Smith, and H.M. Zhang. Distributed vehicular traffic control and safety applications with vgrid. In *Wireless Hive Networks Conference, 2008. WHNC 2008. IEEE*, pages 1–5, Aug. 2008.

[35] CVIS. Cvis - use cases and system requirements. Technical report, CVIS, Novemeber 2006.

[36] T.-S. Dao, K. Y. K. Leung, C. M. Clark, and J. P. Huissoon. Markov-based lane positioning using intervehicle communication. *Intelligent Transportation Systems, IEEE Transactions on*, 8(4):641–650, December 2007.

[37] Sasan Dashtinezhad, Tamer Nadeem, Bogdan Dorohonceanu, Cristian Borcea, Porlin Kang, and Liviu Iftode. TrafficView: A driver assistant device for traffic monitoring based on car-to-car communication. In *IEEE Vehicular Technology Conference*, Milan, Italy, 2004.

[38] Astrid De Witte, Cathy Macharis, Celine Polain, Thrse Steenberghen, Stefaan Van de Walle, and Pierre Lannoy. The impact of "free" public transport:the case of brussels. *Transportation Research*, 42:671–689, 2006.

[39] Hussein Dia, S. Panwai, N. Boongrapue, Tu Ton, and N. Smith. Comparative evaluation of power-based environmental emissions models. In *Intelligent Transportation Systems Conference, 2006. ITSC '06. IEEE*, pages 1251–1256, 2006.

[40] S.R. Dickey, B. Bougler, Nelson Kretz, P., J.A. D.A., Ko, and J.A. Misener. Intersection signal controllers and vehicle-infrastructure integration: Putting safe intersections to practice. In *Paper presented at the TRB 87th Annual Meeting Compendium of Papers*. PATH VII California, Transportation Research Board, January 2008.

[41] S.R. Dickey, Ching-Ling Huang, and Xu Guan. Field measurements of vehicle to roadside communication performance. In *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, pages 2179–2183, 30 2007-Oct. 3 2007.

[42] S.R. Dickey and J. VanderWerf. Telematics issues in california's vii testbed. *IT - Information Technology*, 50(4):243–252, 2008.

[43] Marios D. Dikaiakos, Saif Iqbal, Tamer Nadeem, and Liviu Iftode. Vitp: an information transfer protocol for vehicular computing. In *VANET '05: Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks*, pages 30–39, New York, NY, USA, 2005. ACM.

[44] Sandor Dornbush and Anupam Joshi. StreetSmart traffic: Discovering and disseminating automobile congestion using VANET's. In *IEEE Vehicular Technology Conference*, Dublin, Ireland, April 2007.

[45] Jie Du and M.J. Barth. Next-generation automated vehicle location systems: Positioning at the lane level. *Intelligent Transportation Systems, IEEE Transactions on*, 9(1):48–57, March 2008.

[46] EEA. Greenhouse gas emission trends and projections in europe 2006. Technical Report 9, European Environment Agency, October 2006.

[47] EEA. Urban sprawl in europe - the ignored challenge. Technical Report 9, European Environmental Agency, November 2006.

[48] D Egan. The emergence of zigbee in building automation and industrial control. *Computing & Control Engineering Journal*, 16:14–19, April-May 2005.

[49] Stephan Eichler. Performance evaluation of the IEEE 802.11p WAVE communication standard. In *66th IEEE Vehicular Technology Conference*, pages 2199–2203, Baltimore, MD, USA, 2007.

[50] F. Elbahhar, A. Rivenq, M. Heddebaut, and J.M. Rouvaen. Using UWB gaussian pulses for inter-vehicle communications. *Communications, IEE Proceedings-*, 152(2):229–234, April 2005.

[51] Dan Empey. Path vehicles will roll at demo 2003. *Intellimotion*, 10(1):8, 2002.

[52] Jakob Eriksson, Hari Balakrishnan, and Samuel Madden. Cabernet: Vehicular Content Delivery Using WiFi. In *14th ACM MOBICOM*, San Francisco, CA, September 2008.

[53] Jakob Eriksson, Lewis Girod, Bret Hull, Ryan Newton, Samuel Madden, and Hari Balakrishnan. The Pothole Patrol: Using a Mobile Sensor Network for Road Surface Monitoring. In *The Sixth Annual International conference on Mobile Systems, Applications and Services (MobiSys 2008)*, Breckenridge, U.S.A., June 2008.

[54] J.R. Evans and E. Minieka. *Optimization Algorithms for Networks and Graphs*, chapter 4, pages 109–112. Dekker, New York, 1992.

[55] FCC. Fcc allocates spectrum in 5.9 ghz range for intelligent transportation systems uses. Press Release, Oct 1999.

[56] A. Festag, G. Noecker, M. Strassberger, A. Lbke, B. Bochow, M. Torrent-Moreno, S. Schnaufer, R. Eigner, C. Catrinescu, and J. Kunisch. Now - network on wheels: Project objectives, technology and achievements. In *6th International Workshop on Intelligent Transportation (WIT)*, March 2008.

[57] Marco Fiore, Jerome Harri, Fethi Filali, and Christian Bonnet. Vehicular mobility simulation for vanets. In *ANSS '07: Proceedings of the 40th Annual Simulation Symposium*, pages 301–309, Washington, DC, USA, 2007. IEEE Computer Society.

[58] M.D. Dikaiakos A. Florides, T. Nadeem, and L. Iftode. Location-aware services over vehicular ad-hoc networks using car-to-car communication. *Selected Areas in Communications, IEEE Journal on*, 25(8):1590–1602, October 2007.

[59] Transport for London. Central london congetion charging: Impacts monitoring, sixth annual report. Technical report, Transport for London, July 2008.

[60] H. Fujii, O. Hayashi, and N. Nakagata. Experimental research on inter-vehicle communication using infrared rays. In *Intelligent Vehicles Symposium, 1996., Proceedings of the 1996 IEEE*, pages 266–271, Sep 1996.

[61] J. Gibbons and S. Ruth. Municipal wi-fi: big wave or wipeout? *Internet Computing, IEEE*, 10(3):66–71, May-June 2006.

[62] Norbert Goebel, Jedrzej Rybicki, Bjrn Scheuermann, and Martin Mauve. A Peer-to-Peer-based Traffic Information System. In *MobiCom 2009: Fifteenth ACM SIGMOBILE International Conference on Mobile Computing and Networking, Demo Track*, September 2009.

[63] A.V. Goldberg and M.O. Ball. Computing the shortest path: A* search meets graph theory. Technical Report MSR-TR-2004-24, Microsoft Research,2004.2004.

[64] Victor Gradinescu, Cristian Gorgorin, Raluca Diaconescu, Valentin Cristea, and Liviu Iftode. Adaptive traffic lights using car-to-car communication. In *IEEE Vehicular Technology Conference*, Dublin, Ireland, April 2007.

[65] Ann Guy and Wei-Bin Zhang. Next step in vehicle assist and automation. *Intellimotion*, 15:1–2, 9–10, 2009.

[66] J. Härri, F. Filali, C. Bonnet, and Marco Fiore. Vanetmobisim: generating realistic mobility patterns for vanets. In *VANET '06: Proceedings of the 3rd international workshop on Vehicular ad hoc networks*, pages 96–97, New York, NY, USA, 2006. ACM.

[67] Mark Hartong, Rajini Goel, Csilla Farkas, and Duminda Wijesekera. PTC-VANET interactions to prevent highway rail intersection crossing accidents. In *IEEE Vehicular Technology Conference*, Dublin, Ireland, April 2007.

[68] Hong-Yu Huang, Pei-En Luo, Minglu Li, Da Li, Xu Li, Wei Shu, and Min-You Wu. Performance evaluation of suvnet with real-time traffic data. *Vehicular Technology, IEEE Transactions on*, 56(6):3381–3396, Nov. 2007.

[69] Jesse Hurdus, Andrew Bacha, Cheryl Bauman, Stephen Cacciola, Ruel Faruque, Peter King, Chris Terwelp, Patrick Currier, Dennis Hong, Alfred Wicks, and Charles Reinholtz. Victortango architecture for autonomous navigation in the darpa urban challenge. *JOURNAL OF AEROSPACE COMPUTING, INFORMATION, AND COMMUNICATION*, 5:506–529, December 2008.

[70] IEEE. Trial-use standard for WAVE - multi-channel operation. *IEEE Std 1609.4-2006*, pages 1–74, November 2006.

[71] IEEE. Trial-use standard for WAVE - resource manager. *IEEE Std 1609.1-2006*, pages c1–63, Oct 2006.

[72] IEEE. Trial-use standard for WAVE - security services for applications and management messages. *IEEE Std 1609.2-2006*, June 2006.

[73] IEEE. IEEE P802.11p/D3.0, Draft Amendment for Wireless Access in Vehicular Environments (WAVE). July 2007.

[74] IEEE. Trial-use standard for WAVE networking services. *IEEE Std 1609.3-2007*, pages 1–87, April 2007.

[75] L. Iftode, S. Smaldone, M. Gerla, and J. Misener. Active highways (position paper). In *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, pages 1–5, Sept. 2008.

[76] Y. Inoue and M. Nakagawa. Mac protocol for inter-vehicle communication network using spread spectrum technique. In *Vehicle Navigation and Information Systems Conference, 1994. Proceedings., 1994*, pages 149–152, Aug-2 Sep 1994.

[77] ISO. raffic and travel information (tti) - tti via transport protocol expert group (tpeg) data-streams – part 6: Location referencing applications. *ISO TS 18234-6:2006*, 2006.

[78] ISO. Traffic and travel information (tti) – tti via transport protocol expert group (tpeg) data-streams – part 1: Introduction, numbering and versions. *ISO TS 18234-1:2006*, 2006.

[79] ISO. Traffic and travel information (tti) – tti via transport protocol expert group (tpeg) data-streams – part 2: Syntax, semantics and framing structure (ssf). *ISO TS 18234-2:2006*, 2006.

[80] ISO. Traffic and travel information (tti) – tti via transport protocol expert group (tpeg) data-streams – part 3: Service and network information (sni) application. *ISO TS 18234-3:2006*, 2006.

[81] ISO. Traffic and travel information (tti) – tti via transport protocol expert group (tpeg) data-streams – part 4: Road traffic message (rtm) application. *ISO TS 18234-4:2006*, 2006.

[82] ISO. Traffic and travel information (tti) – tti via transport protocol expert group (tpeg) data-streams – part 5: Public transport information (pti) application. *ISO TS 18234-5:2006*, 2006.

[83] ISO. Traffic and travel information (tti) – tti via transport protocol experts group (tpeg) extensible markup language (xml) – part 1: Introduction, common data types and tpegml. *ISO TS 24530-1:2006*, 2006.

[84] ISO. Traffic and travel information (tti) – tti via transport protocol experts group (tpeg) extensible markup language (xml) – part 2: tpeg-locml. *ISO TS 24530-2:2006*, 2006.

[85] ISO. Traffic and travel information (tti) – tti via transport protocol experts group (tpeg) extensible markup language (xml) – part 3: tpeg-rtmml. *ISO TS 24530-3:2006*, 2006.

[86] ISO. Traffic and travel information (tti) – tti via transport protocol experts group (tpeg) extensible markup language (xml) – part 4: tpeg-ptiml. *ISO TS 24530-4:2006*, 2006.

[87] Joel VanderWerf James A. Misener, Susan Dickey and Raja Sengutpta. *Vehicle Infastructure Cooperation*, chapter 3, pages 3–1 to 3–35. Chapman & Hall/CRC, 2009.

[88] D. Jiang and L. Delgrossi. IEEE 802.11p: Towards an international standard for wireless access in vehicular environments. pages 2036–2040, May 2008.

[89] Daniel Jiang, Qi Chen, and Luca Delgrossi. Optimal data rate selection for vehicle safety communications. In *VANET '08: Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking*, pages 30–38, New York, NY, USA, 2008. ACM.

[90] B. Jones. DSRC-Linking the Vehicle and the Road. In *VII Public Meeting, ITS America. San Francisco, California*, 2005.

[91] Takayoshi Kagei. New advances in electronic toll collection (etc) systems. In *15th World Congress on ITS*. Japanese Ministry of Land, Infrastructure and Transport, November 2008.

[92] J. Kaltwasser and J. Kassubek. A new cooperative optimized channel access for inter-vehicle communication. In *Vehicle Navigation and Information Systems Conference, 1994. Proceedings., 1994*, pages 145–148, Aug-2 Sep 1994.

[93] F.K. Karnadi, Zhi Hai Mo, and Kun chan Lan. Rapid generation of realistic mobility models for vanet. *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, pages 2506–2511, March 2007.

[94] Brad Karp and H.T. Kung. GPSR: Greedy perimeter stateless routing for wireless networks. In *MOBICOM 2000*, pages 243 – 254, Boston, MA, USA, 2000.

[95] ZuWhan Kim. Robust lane detection and tracking in challenging scenarios. *Intelligent Transportation Systems, IEEE Transactions on*, 9(1):16–26, March 2008.

[96] Tom Kloster, James Daisa, and Rich Ledbetter. Linking land use and transportation through street design. In *Urban Street Symposium*, Dallas, Texas, June 1999.

[97] Timo Kosch, Ilse Kulp, Marc Bechler, Markus Strassberger, Benjamin Weyl, and Robert Lasowski. Communication architecture for cooperative systems in europe. *IEEE Communications Magazine*, 47:116–125, May 2009.

[98] Stefan Krauß. *Microscopic Modelling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics*. PhD thesis, 1998.

[99] J.S. Kwak and Jae Hong Lee. Infrared transmission for intervehicle ranging and vehicle-to-roadside communication systems using spread-spectrum technique. *Intelligent Transportation Systems, IEEE Transactions on*, 5(1):12–19, March 2004.

[100] J. LeBrun, J. Anda, C.N. Chuah, M. Zhang, and D. Ghosal. VGrid: Vehicular ad hoc networking and computing grid for intelligent traffic control. In *61st IEEE Vehicular Technology Conference*, pages 2905–2909, 2005.

[101] John J. Leonard, Jonathan P. How, Seth J. Teller, Mitch Berger, Stefan Campbell, Gaston A. Fiore, Luke Fletcher, Emilio Frazzoli, Albert S. Huang, Sertac Karaman, Olivier Koch, Yoshiaki Kuwata, David Moore, Edwin Olson, Steve Peters, Justin Teo, Robert Truax, Matthew Walter, David Barrett, Alexander Epstein, Keoni Maheloni, Katy Moyer, Troy Jones, Ryan Buckley, Matthew E. Antone, Robert Galejs, Siddhartha Krishnamurthy, and Jonathan Williams. A perception-driven autonomous urban vehicle. *J. Field Robotics*, 25(10):727–774, 2008.

[102] Da Li, Hongyu Huang, Xu Li, Minglu Li, and Feilong Tang. A distance-based directional broadcast protocol for urban vehicular ad hoc network. *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*, pages 1520–1523, Sept. 2007.

[103] Tim Lomax and David Schrank. Urban mobility study. Technical report, Texas Transportation Institute, 2005.

[104] Cathy Macharis, Astrid De Witte, Celine Polain, Thrse Steenberghen, Stefaan Van de Walle, and Pierre Lannoy. Impact and assessment of "free" public transport measures: lessons from the case study of brussels. *European Transport: Special Issue on pricing and subsidies in transport*, pages 26–46, 2006.

[105] D Mahalel and A S Hakkert. Traffic arrival patterns on a cross section of a multilane highway. *Transportation Research Part A: General*, 17(4):263–270, 1983.

[106] Rahul Mangharam, Daniel Weller, Raj Rajkumar, Priyantha Mudalige, and Fan Bai. Groovenet: A hybrid simulator for vehicle-to-vehicle networks. In *Mobile and Ubiquitous Systems - Workshops, 2006. 3rd Annual International Conference on*, pages 1–8, July 2006. Invited Paper.

[107] A. Mann and J. Ruckert. A new concurrent slot assignment protocol for traffic information exchange. In *Vehicular Technology Conference, 1988, IEEE 38th*, pages 503–508, Jun 1988.

[108] Patrice Marcotte and Michael Patriksson. Chapter 10 traffic equilibrium. *Handbooks in Operations Research and Management Science*, 14:623–713, 2007.

[109] Takuya Maruyama and Noboru Harata. Difference between area-based and cordon-based congestion pricing: Investigation by trip-chain-based network equilibrium model with nonadditive path costs. *Transportation Research Record*, 1964:1–8, 2006.

[110] R. Mathar and W. Kremer. Histogram and integral based evaluation of an experimental inter vehicle communication trial. In *Vehicular Technology Conference, 1993 IEEE 43rd*, pages 313–317, May 1993.

[111] R. Mathar and A. Mann. Analyzing a distributed slot assignment protocol by markov chains. In *Vehicular Technology Conference, 1992, IEEE 42nd*, pages 715–720 vol.2, May 1992.

[112] Ian Mcneill. *An Encyclopaedia of the History of Technology*. Routledge, 1990.

[113] Isaac Miller, Mark Campbell, Dan Huttenlocher, Frank-Robert Kline, Aaron Nathan, Sergei Lupashin, Jason Catlin, Brian Schimpf, Pete Moran, Noah Zych, Ephrahim Garcia, Mike Kurdziel, and Hikaru Fujishima. Team cornell's skynet: Robust perception and planning in an urban environment. *J. Field Robotics*, 25(8):493–527, 2008.

[114] J. Miller. *Transportation on College and University Campuses*, chapter Chapter 5: Case Studies, pages 41–46. National Academy Press, 2001.

[115] Kiyoshi Mizui, Masatoshi Uchida, and Masao Nakagawa. Vehicle-to-vehicle communications and ranging system using spread spectrum techniques. *Electronics and Communications in Japan (Part I: Communications)*, 79(12):106–115, 1996.

[116] M. Montemerlo, J. Becker, S. Bhat, H. Dahlkamp, D. Dolgov, S. Ettinger, D. Haehnel, T. Hilden, G. Hoffmann, B. Huhnke, D. Johnston, S. Klumpp, D. Langer, A. Levandowski, J. Levinson, J. Marcil, D. Orenstein, J. Paefgen, I. Penny, A. Petrovskaya, M. Pflueger, G. Stanek, D. Stavens, A. Vogt, and S. Thrun. Junior: The stanford entry in the urban challenge. *Journal of Field Robotics*, 25:569–597, 2008.

[117] James Morris. Tomtom hd traffic. *Trusted Reviews*, June 2009.

[118] S. Munder, C. Schnorr, and D.M. Gavrila. "pedestrian detection and tracking using a mixture of view-based shape-texture models". *Intelligent Transportation Systems, IEEE Transactions on*, 9(2):333–343, June 2008.

[119] Tamer Nadeem, Pravin Shankar, and Liviu Iftode. A comparative study of data dissemination models for vanets. In *Mobile and Ubiquitous Systems: Networking & Services, 2006 Third Annual International Conference on*, pages 1–10, July 2006.

[120] Kai Nagel and Michael Schreckenberg. A cellular automaton model for freeway traffic. *Journal de Physique I France*, 2:2221–2229, September 1992.

[121] A O'Brien. New ramp metering algorithm improves systemwide travel time. *TR News*, 209:38–39, August 2000.

[122] National Academy of Sciences. The global positioning system: The path from research to human benefit (beyond discovery series). *The National Academies Press*, 1996.

[123] Tadashi Okutani. Policy for cooperative its in japanese smartway project. In *15th World Congress on ITS*. Japanese Ministry of Land, Infrastructure and Transport, November 2008.

[124] Stephan Olariu and Michele C. Weigle, editors. *Vehicular Networks: From Theory to Practice*. Chapman & Hall/CRC, 2009.

[125] John S. Otto and Fabin E. Bustamante. Distributed or centralized traffic advisory systemsthe applications take. In *IEEE SECON 2009*, June 2009.

[126] M Papageorgiou, H Hadj-Salem, and J-M Blosseville. Alinea: A local feedback control law for on-ramp metering. *Transportation Research Record*, 1320:58–64, 1991.

[127] Michal Piorkowski, Maxim Raya, Ada Lugo, Panos Papadimitratos, Matthias Grossglauser, and Jean-Pierre Hubaux. TraNS: Joint Traffic and Network Simulator for VANETs. In *MobiCom 2007*, 2007.

[128] Michal Piorkowski, Maxim Raya, Ada Lugo, Panos Papadimitratos, Matthias Grossglauser, and Jean-Pierre Hubaux. TraNS: Realistic Joint Traffic and Network Simulator for VANETs. *ACM SIGMOBILE Mobile Computing and Communications Review*, January 2008.

[129] EBU B/TPEG Project. Guidelines for tpeg in dab. Technical report, EBU, Augyst 2000.

[130] EBU B/TPEG Project. Guidelines for tpeg on the internet. Technical report, EBU, December 2002.

[131] PTV-AG. VISSIM. http://www.english.ptv.de/cgi-bin/traffic/traf_vissim.pl.

[132] C.M. Roberts. Radio frequency identification (rfid). *Computers & Security*, 25(1):18 – 26, 2006.

[133] Jedrzej Rybicki, Bjrn Scheuermann, Wolfgang Kiess, Christian Lochert, Pezhman Fallahi, and Martin Mauve. Challenge: Peers on Wheels - A Road to New Traffic Information Systems. In *MobiCom 2007: Proceedings of the 13th Annual International Conference on Mobile Computing and Networking*, pages 215–221, September 2007.

[134] Jedrzej Rybicki, Bjrn Scheuermann, Markus Koegel, and Martin Mauve. PeerTIS - A Peer-to-Peer Traffic Information System. In *VANET 2009: Proceedings of the Sixth ACM International Workshop on Vehicular Inter-Networking*, September 2009.

[135] Jos Santa, Antonio F. Gmez-Skarmeta, and Marc Snchez-Artigas. Architecture and evaluation of a unified v2v and v2i communication system based on cellular net-

works. *Computer Communications*, 31(12):2850 – 2861, 2008. Mobility Protocols for ITS/VANET.

[136] Jos Santa, Andrs Muoz, and Antonio F G Skarmeta. *A Context-Aware Solution for Personalized En-route Information Through a P2P Agent-Based Architecture*, volume 4707 of *Lecture Notes in Computer Science*. Springer, 2007.

[137] Jose Santa, Rafael Toledo-Moreo, and Antonio F. Gomez-Skarmeta. A novel vehicle communication paradigm based on cellular networks for improving the safety in roads. *International Journal of Intelligent Information and Database Systems*, 2(2):240 – 257, 2008.

[138] Georgina Santos and Blake Shaffer. Preliminary results of the london congestion charging scheme. *Public Works Management & Policy*, 9(2):164–181, 2004.

[139] Dieter Seeberger. Standardisation overview. Technical report, COMeSafety, August 2006.

[140] Guna Seetharaman, Arun Lakhotia, and Erik Philip Blasch. Unmanned vehicles come of age: The DARPA grand challenge. *IEEE Computer Magazine*, 39(12):26–29, Dec. 2006.

[141] S.E. Shladover. Path at 20history and major milestones. *Intelligent Transportation Systems, IEEE Transactions on*, 8(4):584–592, Dec. 2007.

[142] S.E. Shladover, C.A. Desoer, J.K. Hedrick, M. Tomizuka, J. Walrand, W.-B. Zhang, D.H. McMahon, H. Peng, S. Sheikholeslam, and N. McKeown. Automated vehicle control developments in the path program. *Vehicular Technology, IEEE Transactions on*, 40(1):114–130, Feb 1991.

[143] M.L. Sichitiu and M. Kihl. Inter-vehicle communication systems: a survey. *Communications Surveys & Tutorials, IEEE*, 10(2):88–105, Quarter 2008.

[144] I. Skog and P. Handel. In-car positioning and navigation technologiesa survey. *Intelligent Transportation Systems, IEEE Transactions on*, 10(1):4–21, March 2009.

[145] Stevie Smith. Vexia econav saves money and the environment. *The Tech Herald*, Feb 2009.

[146] Christoph Sommer, Zheng Yao, Reinhard German, and Falko Dressler. Simulating the influence of ivc on road traffic using biderectionally coupled simulators. In *27th IEEE Conference on Computer Communications (IEEE INFOCOM 2008): Mobile Networking for Vehicular Environments (MOVE 2008)*, Phoenix, AZ, USA, April 2008.

[147] H. Soroush, N. Banerjee, M. D. Corner, B. N. Levine, and B. Lynn. Dome: A diverse outdoor mobile testbed. Dept. of computer science technical report um-cs-2009-23, Univ. of Massachusetts Amherst, May 2009.

[148] Hamed Soroush, Nilanjan Banerjee, Aruna Balasubramanian, Mark D. Corner, Brian Neil Levine, and Brian Lynn. DOME: A Diverse Outdoor Mobile Testbed. In *Proc. ACM Intl. Workshop on Hot Topics of Planet-Scale Mobility Measurements (HotPlanet)*, June 2009.

[149] Dominic Stead and Harry Geerlings. Integrating transport, land use planning and environment policy. *Innovation: The European Journal of Social Science Research*, 18(4):443–453, December 2005.

[150] A. Sugiura and C. Dermawan. In traffic jam ivc-rvc system for its using bluetooth. *Intelligent Transportation Systems, IEEE Transactions on*, 6(3):302–313, Sept. 2005.

[151] T. Taleb, K. Ooi, and K. Hashimoto. An efficient collision avoidance strategy for its systems. In *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, pages 2212–2217, 31 2008-April 3 2008.

[152] Han-Shue Tan. An automated snowblower for highway winter operations. *Intellimotion*, 10(4):1, 6–9, 2004.

[153] Han-Shue Tan, Fanping Bu, Scott Johnston, Benedicte Bougler, Wei-Bin Zhang, and Sonja Sun. Field demonstration and tests of lane assist/guidance and precision docking technology. Technical report, California PATH, January 2009.

[154] Brian D. Taylor. Rethinking traffic congestion. *Access*, 21:8–16, October 2002.

[155] Telecompaper. Garmin to offer nuvi 1690 on at&t network. *Telecom Paper*, September 2009.

[156] Telecompaper. Tomtom signs up at&t for live services in us. *Telecom Paper*, September 2009.

[157] Arvind Thiagarajan, Lenin Ravindranath, Katrina LaCurts, Samuel Madden, Hari Balakrishnan, Sivan Toledo, and Jakob Eriksson. Vtrack: accurate, energy-aware road traffic delay estimation using mobile phones. In *SenSys '09: Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, pages 85–98, New York, NY, USA, 2009. ACM.

[158] Marc Torrent-Moreno, Andreas Festag, and Hannes Hartenstein. System design for information dissemination in VANETs. In *3rd International Workshop on Intelligent Transportation*, Hamburg, Germany, March 2006.

[159] Martin Treiber, Ansgar Hennecke, and Dirk Helbing. Congested traffic states in empirical observations and microscopic simulations. *Physical Review E*, 62:1805, 2000.

[160] Understanding-Science. Ozone depletion: Uncovering the hidden hazard of hairspray. Online.

[161] UNFPA. State of world population 2007: Unleashing the potential of urban growth. Technical report, United Nations Population Fund, 2007.

[162] Christopher Urmson, Joshua Anhalt, Hong Bae, James (Drew) Bagnell, Christopher Baker, Robert E Bittner, Thomas Brown, M. N. Clark, Michael Darms, Daniel

Demitrish, John Dolan, David Duggins, David Ferguson, Tugrul Galatali, Christopher M Geyer, Michele Gittleman, Sam Harbaugh, Martial Hebert, Thomas Howard, Sascha Kolski, Maxim Likhachev, Bakhtiar Litkouhi, Alonzo Kelly, Matthew McNaughton, Nick Miller, Jim Nickolaou, Kevin Peterson, Brian Pilnick, Ragunathan Rajkumar, Paul Rybski, Varsha Sadekar, Bryan Salesky, Young-Woo Seo, Sanjiv Singh, Jarrod M Snider, Joshua C Struble, Anthony (Tony) Stentz, Michael Taylor, William Red L. Whittaker, Ziv Wolkowicki, Wende Zhang, and Jason Ziglar. Autonomous driving in urban environments: Boss and the urban challenge. *Journal of Field Robotics Special Issue on the 2007 DARPA Urban Challenge, Part I*, 25(8):425–466, June 2008.

[163] M. van den Berg, T. Bellemans, B. De Schutter, B. De Moor, and H. Hellendoorn. Control of traffic with anticipative ramp metering. In *In Proc. 84th Annual Meeting of the Transportation Research Board*, Washington, DC, 2005.

[164] P. Varaiya. Smart cars on smart roads: problems of control. *Automatic Control, IEEE Transactions on*, 38(2):195–207, Feb 1993.

[165] R. Verdone. Multihop r-aloha for intervehicle communications at millimeter waves. *Vehicular Technology, IEEE Transactions on*, 46(4):992–1005, Nov 1997.

[166] Vivid Solutions. *JUMP Developer Guide*, 2003.

[167] J. Voelcker. Cars get street smart: No thanks to DARPA. *Spectrum, IEEE*, 44(10):16–18, October 2007.

[168] J. Voelcker. Autonomous vehicles complete darpa urban challenge. *Spectrum, IEEE*, Online, November 2008.

[169] Manos VOUGIOUKAS. Implementing fair and efficient pricing and subsidy policies in urban transport. In *ECMT/OECD Workshop*, Athens, Greece, 1999.

[170] Paul Waddell. Urbansim: Modeling urban development for land use, transportation and environmental planning. *Journal of the American Planning Association*, 68(3):297–314, 2002.

[171] Shie-Yuan Wang, Yow-Wei Cheng, Chih-Che Lin, Wei-Jyun Hong, and Ting-Wei He. A vehicle collision warning system employing vehicle-to-infrastructure communications. In *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, pages 3075–3080, 31 2008-April 3 2008.

[172] S.Y. Wang, C.L. Chou, Y.H. Chiu, Y.S. Tzeng, M.S. Hsu, Y.W. Cheng, W.L. Liu, and T.W. Ho. Nctuns 4.0: An integrated simulation platform for vehicular traffic, communication, and network researches. In *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, pages 2081–2085, 30 2007-Oct. 3 2007.

[173] S.Y. Wang, C.L. Chou, and C.C. Lin. Increasing wide-area download throughputs on the roads by trunking multiple cellular channels over a vehicular ad hoc network. *Computer Communications*, 32(2):268 – 280, 2009.

[174] Yi Wang, A. Ahmed, B. Krishnamachari, and K. Psounis. IEEE 802.11p performance evaluation and protocol enhancement. In *Vehicular Electronics and Safety, 2008. ICVES 2008. IEEE International Conference on*, pages 317–322, Sept. 2008.

[175] Michael Wegener and Franz Fuerst. Land-use transport interaction: State of the art. Urban/Regional 0409005, EconWPA, September 2004. available at http://ideas.repec.org/p/wpa/wuwpur/0409005.html.

[176] C. Wewetzer, M. Caliskan, K. Meier, and A. Luebke. Experimental evaluation of umts and wireless lan for inter-vehicle communication. In *Telecommunications, 2007. ITST '07. 7th International Conference on ITS*, pages 1–6, June 2007.

[177] T.L. Willke, P. Tientrakool, and N.F. Maxemchuk. A survey of inter-vehicle communication protocols and their applications. *Communications Surveys & Tutorials, IEEE*, 11(2):3–20, Quarter 2009.

[178] Lars Wischof, Andr Ebner, and Hermann Rohling. Information dissemination in self organising intervehicle networks. *Intelligent Transportation Systems, IEEE Transactions on*, 6(1):90–101, March 2005.

[179] Gongjun Yan, Michele C. Weigle, and Stephan Olariu. A novel parking service using wireless networks. In *Proceedings of the IEEE International Conference on Service Operations, Logistics and Informatics*, Chicago, jul 2009. Best Student Paper Award.

[180] Lin Yang, Jinhua Guo, and Ying Wu. Piggyback cooperative repetition for reliable broadcasting of safety messages in vanets. In *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE*, pages 1–5, Jan. 2009.

[181] T. Yashiro, T. Kondo, H. Yagome, M. Higuchi, and Y. Matsushita. A network based on inter-vehicle communication. In *Intelligent Vehicles '93 Symposium*, pages 345–350, Jul 1993.

[182] W. Zhang, A. Festag, R. Baldessari, and L. Le. Congestion control for safety messages in vanets: Concepts and framework. In *ITS Telecommunications, 2008. ITST 2008. 8th International Conference on*, pages 199–203, Oct. 2008.