Context-Aware Aided Parking Solutions Based on VANET

PhD Thesis

Abdulmalik A. Alhammad

This thesis is submitted in partial fulfillment of the requirements

for the degree of Doctor of Philosophy

Software Technology Research Laboratory

De Montfort University

Leicester - United Kingdom

2014

Dedication

To my father

Abdulrahman

You have always been my motivation

Thank you for all the sacrifice you have made and continue to make for me

To my mother

Sultanah

For all your prayers, unconditional love and for believing in me

Thank you for all that you have done for me

To my wife

Hind

For your endless love, support and patience

Thank you for being there, especially in the trying times

Abstract

Vehicular Ad-hoc Network (VANET) is a special application of the Mobile Ad-hoc Network (MANET) for managing road traffic and substantially contributes to the development of Intelligent Transportation Systems (ITS). VANET was introduced as a standard for data communication between moving vehicles with and without fixed infrastructure. It aims to support drivers by improving safety and driving comfort as a step towards constructing a safer, cleaner and a more intelligent environment. Nowadays, vehicles are manufactured equipped with a number of sensors and devices called On Board Units (OBU) assisting the vehicle to sense the surrounding environment and then process the context information to effectively manage communication with the surrounding vehicles and the associated infrastructure.

A number of challenges have emerged in VANET that have encouraged researchers to investigate this concept further. Many of the recent studies have applied different technologies for intelligent parking management. However, despite all the technological advances, researchers are no closer to developing a system that enables drivers to easily locate and reserve a parking space. Limited resources such as energy, storage space, availability and reliability are factors which could have contributed to the lack success and progress in this area.

The task then is to close these gaps and present a novel solution for parking.

This research intends to address this need by developing a novel architecture for locating and reserving a parking space that best matches the driver's preferences and vehicle profile without distracting the driver.

The simple and easy-to-use mechanism focuses on the domain of an intelligent parking system that exploits the concept of InfoStation (IS) and context-aware system creating a single framework to locate and reserve a parking space. A three tier network topology comprising of vehicles, IS and the InfoStation Centre (ISC) has been proposed as the foundation of the on-street parking system architecture. The thesis attempts to develop the architecture of a parking management solution as a comfort-enhancing application that offers to reduce congestion related stress and improve the driver experience by reducing the time it takes to identify and utilise a parking space that is available.

Declaration

I declare that the work described in this thesis is original work undertaken by me for the degree of Doctor of Philosophy, at the software Technology Research Laboratory (STRL), at De Montfort University, United Kingdom.

No part of the material described in this thesis has been submitted for any award of any other degree or qualification in this or any other university or college of advanced education.

This thesis is written by me and produced using LaTeX.

Abdulmalik Alhammad

Publications

A Alhammad, F Siewe and A Al-bayatti, " An Info Station-Based Context-Aware On- Street Parking System." *In proceeding of the IEEE of the International Conference on Computer System and Industrial Informatics (ICCSII),* 2012.

Acknowledgments

It is a great honour to have this opportunity to dedicate my appreciation and acknowledge the contribution of supporters and well-wishers who have helped me to conduct my research and write this PhD thesis.

First and foremost, I offer my thanks and humble words of praise to my Lord, the most merciful ALLAH for showering me with his infinite blessings and resourcing me with the gifts and talents for fulfilling my dreams.

Next, I wish to extend my special and heartfelt appreciation to my Supervisor, Dr. Ali Al-Bayatti, without whom this thesis would not have been possible. I have benefited greatly from his experience, wisdom and knowledge and I am proud to say that I conducted my PhD under his guiding supervision. I would also like to thank Dr. François Siewe for his valuable comments, technical expertise and professional guidance which have greatly improved this thesis.

My love and life to my parents. I am indebted to them both for making me the son and the man that I am. I stand tall today because of their sacrifice, advice and prayers. Thank you and I love you forever and always.

Special thanks go to my lovely and beautiful wife Hind. Her care and attention gave me the strength when I needed it the most. Thank you for being in my life and for sharing the good and bad times.

I would like also to thank my beloved and beautiful daughters Reema and Fajer. I still remember the times when you both filled me with happiness, joy and courage when I looked into your eyes, especially during stressful times.

My thanks and love to my brothers and sisters for their unwavering support and understanding.

I must not forget to acknowledge my dear friend, Dr. Khalid Aldriwish for believing in me and for being a source of encouragement for me throughout my PhD studies. I also want to thank my dear friend and office mate Dr. Moath Al-Doori for all the discussions and laughs we had.

I want to thank all my friends in Saudi Arabia, United Kingdom and everywhere for their friendship and support over the years.

Last but not least, I would like to thank all the members of the STRL department for being a family to me.

Contents

List of Figures

List of Tables

List of Abbreviation

ADAS	Advanced Driver Assistant System
AmI	Ambient Intelligence
ASTM	American National Standard for Metric Practice
AU	Application Unit
BS	Base Stations
CALM	Continuous Air Interface for Long and Medium range
Car2Car CC	Car2Car Communication Consortium
CCA	Calculus of Context-aware Ambients
ccaPL	Calculus of Context-aware Ambients Programming Language
CE	Context Expression
CEPT	European Conference of Postal and Telecommunications Administrations
ECC	Electronic Communications Committee
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
DAB	Digital Audio Broadcasting
DARPA	Defense Advanced Research Projects Agency

DDB	Digital Data Broadcast
DFW	Dallas/Fort Worth International Airport
DoD	Department of Defense
DSRC	Dedicated Short-Range Communications
DVB	Digital Video Broadcasting
DVLA	Driver and Vehicle Licensing Agency
EEBL	Emergency Electronic Brake Light
FCC	Federal Communication Commission
FDM	Frequency Division Multiplexing
FM	Frequency Modulation
GFG/GPSR	Greedy-Face-Greedy / Greedy Perimeter Stateless Routing
GPRS	General Packet Radio Service
GPS	Global Positioning Systems
GUI	Graphical User Interface
HSCSD	High-Speed Circuit-Switched Data
IDAS	Intelligent Driver Assistance system
IEEE	Institute of Electrical and Electronics Engineers
InVC	In-Vehicle Communication
IRM	Image Rejection Mixer
IS	InfoStation

ISC	InfoStation Centre
ISM	Industrial, Scientific and Medical radio bands
ITS	Intelligent Transportation Systems
IVC	Inter-Vehicular Communications
JVM	Java Virtual Machine
LED	Light Emitting Diode
LPGs	Local Peer Groups
MAC	Media Access Control
MANET	Mobile ad hoc Networks
MDR	Master Data Repository
MOST	Media oriented System Transport
NHTSA	National Highway Trafc Safety Administration
NLP	Natural Language Processing
NTIA	National Telecommunications and Information Administration
NV	Navigation System Unit
OBU	On Board Unit
PCN	Post-Crash Notification
PDA	Personal Digital Assistants
PGI	Parking Guidance Information System
POI	Point-Of-Interest

QoS	Quality of Service
R&D	Research and Development
RDS/TMC	Radio Data System / Traffic Message Channel
RFID	Radio-Frequency Identification
RLAN	Remote Local Area Network
RM	Request Message
RSU	Road-Side Unit
RVC	Road-to-Vehicle Communication
SRD/MG	Short Range Device Maintenance Group
SVA	Slow/Stop Vehicle Advisor
TCP/IP	Transmission Control Protocol/Internet Protocol
TPEG	Transport Protocol Experts Group
TV	Television
UI	User Interface
UMTS	Universal Mobile Telecommunications System
UWB	Ultra-Wide Band
UWB	Ultra Wideband
V2I	Vehicle to Infrastructure
V2V	Vehicle to Vehicle
VANET	Vehicular ad hoc Networks

VIC	Vehicle-to-Infrastructure Communication
VII	Vehicle-Infrastructure Integration
VLSI	Very Large Scale Integration
VSCC	Vehicle Safety Communications Consortium
WAVE	Wireless Access in Vehicular Environments
WiMAX	Worldwide Interoperability for Microwave Access
WUSB	Wireless Universal Serial Bus

Chapter 1

Introduction

Objectives:

- To present the research motivation
- To describe the problem
- To enlist the research question and aims
- To state the research contributions
- To outline the thesis structure

1.1 Research Motivations

In recent years, transport has increasingly become a topical political and environmental issue. From designing fuel-efficient vehicles to improving the vast transportation infrastructure, the transport agenda features frequently in the media headlines. One such concern which is high on the agenda is with regards to the desperate need to ease traffic congestion and there are a number of valid reasons for this. The environmental, economic, and social cost of traffic congestion are being realised by both Governments and the people. For instance, it is reported that congestion that is caused by vehicles searching for a parking space contribute to as much as 45% of inner city traffic in large urban areas. A further study by the Texas Transportation Institute's 2007 Urban Mobility Report calculated the extent of congestion in major urban areas in the U.S. In 2005, the report estimated that Americans wasted 4.2 billion hours and 2.9 billion gallons of gasoline while stuck in congestion which represented an economic loss of over $78 billion, an increase of over 70% from 1995. It is the realisation of the global and local impact in light of these statistics that corroborates the significance of the problem that this thesis seeks to address.

These facts are both interesting and alarming in equal measure because access to and usage of motor vehicles is increasing daily in part due to availability and affordability especially in emerging economies, and this trend is likely to increase year on year. In response to these environmental concerns as well as economic opportunities, car manufacturers are designing and equipping vehicles with the latest technologies that can make driving a pleasurable and stress-less experience with minimum cost to the environment. The advent of the 'Smart Vehicle' is one example of such a response. These smart vehicles can be equipped with a vast array of applications ranging from collision avoidance and assisting the driver with easier driving, to predicting and reacting to the road environment thereby reducing the chances of injuries to the driver and other passengers in the car.

The Advanced Driver Assistant System (ADAS) and the ITS have significantly contributed a number of technologies which coordinate the surrounding information, study the traffic situation and provide real-time feedback thereby instantly assisting the driver. For instance, the Intelligent Driver Assistance System (IDAS) developed by Mercedes-Benz effectively utilises radar sensors and stereo cameras to gain information about the surroundings of the car and provide live feedback to the driver [1].

Vehicles these days also come equipped with an On Board Unit (OBU) that provides a user friendly interface to the drivers and enables the vehicle to communicate with other entities. As the hardware size diminishes in size at an exponential speed largely due to the advancement of Very-Large-Scale Integration (VLSI) technology, a new field of computing and software engineering called 'Pervasive Systems' has been coined which aims to move beyond the realms of laptops, PCs and smart phones and into everyday devices such as cars, motorbikes and even coffee mugs that are embedded with technology and connectivity. A particular area of research in these pervasive systems are the 'Context-Aware Systems' where, by the means of various sensors, an object is made to be aware of its surroundings and adapt to changes in its environment [2].

Concurrently, a considerable amount of research has been conducted in the area of Vehicular Ad hoc Networks (VANET) which offers unique characteristics of high mobility and a rapidly changing network topology due to the constant movement of vehicles/cars at a high speed. A network gets established among the vehicles moving along the streets [3]. Also, accurate time synchronisation can be achieved amongst vehicles by using Global Positioning Systems (GPS) that make real-time applications possible.

The relatively recent concept of InfoStation (IS) has shifted the computing trends from desktops and laptops to vehicle OBUs and has also accounted for a downfall in the VANET system by providing an isolated pocket of high bit-rate coverage [4]. IS has been successfully recognised as an infrastructure providing an ubiquitous wireless coverage with reduced system asymmetries, tolerating longer delays and most importantly, utilising an unlicensed frequency band. With a number of vehicles accessing wireless coverage from an IS, a dynamic network of communication-capable vehicles is created and relevant information can be exchanged between mobile and stationary vehicles [4].

Good though all this is, the thesis offers to alleviate a criticism or rather a shortcoming lodged against the developments to date in this area of work. Where previous systems relied exclusively on the driver's interventions to locate and reserve parking, the solution presented here uses the context-aware VANET system to process all the environmental information and driver-preferences to allocate the parking. Removing this chore from the driver adds to the comfort and safety-value of the driving experience.

These exciting developments in Pervasive Systems coupled with the fascination of how the concepts of context-awareness, VANET and IS from vehicles to parking zones can be extended with the possibility of integrating them has been the motivation behind this research. This integration can be appended to existing infrastructures in part or as a whole, owing to their particular characteristics, and design a more efficient and convenient on-street parking-space location and reservation system.

1.2 Problem Description

The undeniable truth is that car parking space is both limited and costly in almost all the major cities of the world. There is a desperate need for innovative parking systems to be designed for meeting parking demand [5] [6] [7]. Many efficient parking systems have been proposed as part of the ITS and smart parking applications the process of finding nearby parking. In response to the challenge of designing intelligent parking location systems, various policies and architectures have been proposed from time to time. A few of these noteworthy architectures have been designed to have vehicles redeployed with a small processor and a short range wireless transceiver [8].

Considered a vitally important process, smart vehicles must be able to identify a vacant parking space (when needed) either with the help of their own computational infrastructure and OBU or with the help of the neighbouring vehicle's computational infrastructure if a strong inter-network between vehicles is being formed by the use of VANET. A major drawback of such parking systems has been that the vehicle has to predetermine the best possible way to reach the parking spots that are reported to be available. The vehicle has to take into consideration the time required to reach any of the available parking spaces or the walking distance if the final destination is not exactly close to the parking spot. The probability that the particular parking spot will still be free once the vehicle shows up at the location also needs to be calculated [9] [10].

The main goal of this research is to develop a robust parking system that can utilise the information extracted by the sensors, the computational infrastructure in the vehicles, the uninhibited wireless coverage provided by IS and a strong inter-network of mobile and stationary vehicles enabled by VANET. The critical function of such parking system design architecture is to locate and reserve a vacant on-street parking space within a reasonable time whilst it is still vacant consequently reducing the traffic congestion which is after all the main purpose of this system.

However, employing a number of concepts and technologies could result in bulky hardware and have a negative impact on the efficient design of such a context-aware on-street parking system. The process of selecting a combination of the above mechanisms will have a crucial impact on the parking system architecture design, thus adequate attention will need to be paid to ensure that the proposed architecture does not become too complex and cumbersome.

This research proposes to design an efficient and effective parking system based on the secure wireless Dedicated Short-Range Communications (DSRC) communication network, IS and sensor communication along with GPS sensors to make parking navigation and location monitoring convenient and easier.

Driven by personal experience and an empathy for thousands of drivers who have the daily unenviable task of searching for half a decent parking space, has led to the proposition of developing an alternative and more efficient system of locating and reserving parking. Using VANET with DSRC has proved to be the first step in this development process.

The work also gains motivation from the 'environmental' agenda in that minimising the time taken to park a vehicle will significantly have a positive impact on the environment by reducing emissions. Finally, previous work done in this field has provided the biggest motivation for taking this up as the focus for research for the thesis.

1.3 The Aims of the Research

Prior to identifying the research aims, the fundamental question which provides the basis of this thesis needs to be posited:

How can the process of allocating and reserving a car parking space be made smarter by utilising the VANET and context aware system?

Although the main thrust of this research seeks to address the above question, a series of sub-questions will need to be answered in order to deal with the subject fully:

- ***How can an effective on-street parking system architecture be designed by utilising the VANET and context aware system?***
- ***What kind of technology and information is needed to allocate and reserve the appropriate parking spot efficiently?***
- ***How can a model be designed to sense, think and act in order to provide an efficient on street parking service?***

Alongside these questions, a range of additional research aims also need to be addressed:

- To increase the system's reliability and connectivity by using the three tier IS network topology, namely the ISC, IS and the vehicle, and provide the driver with up to date information about the on-street parking zones

- To reduce congestion-related stress by utilising the context aware capability which captures information relating to the vehicle and driver profiles, and automatically allocates a suitable parking space in real-time
- To build an architecture utilising VANET, IS and context aware systems that supports an integrated approach.

1.4 Contributions

This research has produced the following key contributions in line with the five key themes in the thesis, namely:

- An IS-based three tier network topology: A three tier network topology comprising of vehicles, IS and the ISC has been proposed as a foundation of the on-street parking system architecture. This network topology offers a stable coverage to ensure the packet forwarding among the parking system by using the VANET V2V and V2I communication.
- A novel context-aware based architecture: This construct supports an integrated approach for utilising VANET, IS and context aware systems in a single framework providing a comprehensive architecture for smart parking system.
- Parking space location and reservation mechanism: The step-by-step parking space locating and reserving algorithm comes with two main functions: one to show the basic mechanism of parking space location and reserving; and the other for the alternative options.
- Formalisation of the model using CCA: The Calculus of Context-Aware Ambients (CCA) is a suitable notation that is used for modelling context aware applications. As the proposed three tier architecture in this thesis is context aware, it has been formalised and modelled in CCA.

- Validating the mechanism using ccaPL: The ccaPL has been selected as the validation tool for on-street parking behaviour in order to show its effectiveness.

1.5 The Structure of the Thesis

The corpus of this thesis is structured in the following way:

Chapter 2

Sets the scene including a background, brief analysis and critical review of the research conducted in the domain of ITS, context aware systems, IS and VANET. The relatively newer concepts imported from software and pervasive computing, such as context-aware systems and IS with specific and potential application to this domain have been explained thoroughly. This serves to underpin the objectives of the research and to synthesise a meaningful and feasible research question.

Chapter 3

This chapter provides the preliminaries and the background of the context-aware systems and the key ideas relating to context modelling in CCA. CCA is explained as a primitive notation for modelling mobility and context-awareness. This chapter defines the ambients used in the thesis and also represents those ambients as an abstraction of a bounded region within which computation takes place.

Chapter 4

This chapter presents a novel three tier architecture of the proposed intelligent parking system based on VANET, IS and the context-aware systems. It also shows the tier-by-tier details of the architecture.

Chapter 5

In this chapter, the proposed parking reservation architecture has been formalised using the CCA notation in order to prove and validate the parking reservation mechanism.

Chapter 6

Here, the ccaPL executable programming language validates the CCA mathematical specification presented in the previous chapter demonstrating the context-aware and mobility capabilities of the system. Three different experiments have been applied to cover the main possible cases of reserving a vacant parking space.

Chapter 7

This chapter concludes the work presented throughout the thesis with the observed results and provides some suggestions for future work.

Chapter 2

Literature Review

Objectives:

- To present an overview of VANET, IS and context-aware systems
- To evaluate the use of VANET, IS and context-aware systems in the design of parking solutions and to identify some existing gaps in research
- To devise a research problem which integrates the concepts of VANET, IS and context-awareness to design a novel intelligent parking architecture.
- To study and critically examine the earlier work done in the design of Intelligent Parking Systems

2.1 Introduction

With the ever-increasing numbers of cars on the road, finding vacant on-street parking has become a considerably difficult task. More often than not, there is usually a vacant parking space available a few metres away or just around the corner, but there is no means by which the driver can become aware of it [11] [12]. Not having this useful information at hand places the driver at a considerable disadvantage in terms of wasting time, fuel and effort; the resulting frustration has led to the phenomenon commonly known as 'driver's frustration'[13]. The main cause of this waste can be attributed to the lack or non-awareness of available parking spaces nearby and the inability to access this valuable information in real-time. These challenges present an opportunity worth pursuing for researchers to develop a simple, yet effective method that would ease the process of locating and reserving a vacant parking spot [14] [15].

This thesis proposes a design that delivers a more efficient parking system by incorporating the cutting-edge concepts of context-aware systems, IS and VANET in the field of ITS. This chapter provides an extensive examination of the literature available in the area of ITS, and a review of the work which has contributed so far to design more efficient parking systems. A critical review of the existing literature on the application of VANET, IS and context aware systems in designing ITS is evaluated that helps to identify a few areas which have a potential to be used in designing an intelligent parking system. The projects undertaken by various researchers seeking to capitalize on either one or a combination of these concepts to design intelligent parking systems have been studied and their relative pros and cons have been discussed. The state-of-the- art trends and methods adopted in designing context aware parking systems, with or without the use of the concepts of VANET and IS have been formulated into a novel concept of IS based context aware on-street parking system.

The following section introduces a general overview of the three concepts: VANET, IS and context-aware systems. This overview provides an understanding of the basic functioning and applications of these concepts and help to relate them to the design of smart parking systems. The next section highlights the earlier research carried out in the design of smart parking, with a critical analysis of the role and application of these three concepts. This section presents both the achieved success and the encountered limitations that these theories have put forward. The section thereafter synthesises the arguments presented thus far into a novel research question, with the findings developing into a research proposal. This section also presents a few difficulties and limitations that need to be tackled during the research process. The chapter concludes with a summary that covers the findings of the review in a nutshell.

2.2 Background

As mentioned in the introduction, this thesis is a study about designing parking systems that are intelligent enough to communicate with the vehicle, and help a driver to locate vacant parking. This section introduces the concepts of (1)VANET, (2)IS and (3)context aware systems briefly, looking firstly at why these concepts are of importance in the domain of parking design. It then considers these three notions in more depth with an analysis of their application by previous researchers in designing parking systems. The final part of this section tries to build a relationship between these three concepts and to design a proposal to integrate them into a single framework.

2.3 Overview of the Vehicular Ad-hoc Network (VANET)

VANET is currently a widely talked-about topic in the specialist area of wireless communication and is a subset of Mobile Ad hoc Network(MANET) [13] [16] where nodes represent vehicles travelling at high velocity with traffic-determined regularity [17]. This technology which enables communication between vehicles and nearby road-side infrastructure, is only possible because of the wireless sensing device installed in the vehicles [18]. New ventures and associated technologies such as applications for congestion alerts, accident control and weather updates have emerged with the inception of VANET [17]. Simulation is an automated tool which can replicate and reproduce a similar life-like real world result, thus making it possible to test VANET networks to the finest detail before implementation. It is essential to create an authentic mobility model that is as realistic as ad-hoc network communication in order to obtain good results from VANET simulation [19].

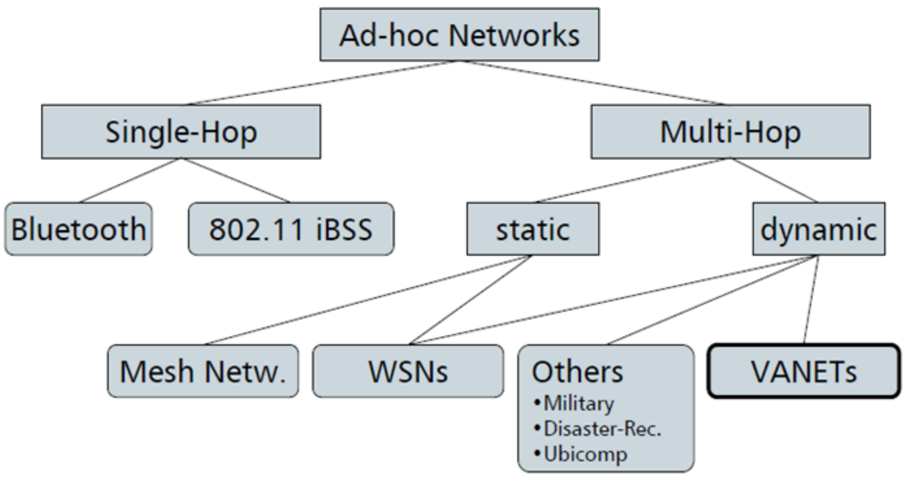


Figure 2.1: Ad-hoc Network Classification

The comfort applications of VANET are aimed at making travelling by a motor vehicle a more pleasant experience for the driver. Part of this experience is to enable access to information relating to parking space availability which has now become one of the most fascinating comfort applications and a significant area of research in VANET. Authors [5], [20], [21], [22] and [8] have proposed some interesting techniques that use sensor communication and GPS communication to locate available parking space at the desired destination. These approaches are implemented using WiMAX and other wireless communication standards. Although most of these approaches have been proposed for parking lots, similar techniques can be applied to on-street parking zones.

2.3.1 History of VANET

In 1998, a team of engineers from Delphi Delco Electronics Systems and IBM Corporation first proposed the concept of a network vehicle [23]. Several recent research projects worldwide have been initiated as a result of the promise of wireless communication to support vehicle safety applications. Such projects include the Vehicle Safety Communications Consortium developing the Dedicated Short-Range Communications (DSRC) technology (USA), the PReVENT project (Europe), the Internet ITS Consortium (Japan), among others [24] [25].

The purpose of the DSRC standard (ASTM E 2213-03) is to provide a means of wireless communication for transportation applications within 1000m at typical road speeds. It was adopted by ASTM and IEEE to cater for emerging wireless vehicle communication needs. DSRC provides seven 10 MHz channels at the 5.9 GHz licensed band for ITS applications. Different channels are designated for different applications, including a reserved channel for vehicle-to-vehicle communications [26]

[27].

2.3.2 VANET Components

VANET enables the vehicles and the Infrastructure to communicate with the help of two basic hardware components: the On-Board Unit (OBU) and the Road-Side Unit (RSU). Nowadays, modern vehicles contain in-built computer-controlled devices and advanced wireless radio transceivers to allow quick and instant message exchange irrespective of the distance between the communicating nodes. The RSU allows the driver to access the required information for example, to find the best route to a destination [25].

2.3.2.1 On-board unit

An OBU is a communication capable hardware component in the vehicle equipped with a Graphical User Interface (GUI). The OBU supports both wireless and wired communication. The OBU works in unison with an Application Unit (AU) which is a tool processing a single or a set of applications whilst exploiting the use of the OBU's communication capacities. An AU can either be permanently connected to an OBU as an integrated part of a vehicle or can be a portable device such as a laptop or PDA that can attach to (and detach from) an OBU. Although wireless connection using Bluetooth, Wireless Universal Serial Bus (WUSB), or Ultra Wideband (UWB) is possible, the AU and OBU are usually connected with a wire-line connection. Although this distinction between AU and OBU is elementary, they can nevertheless reside integrated in a single physical unit [28].

The ad hoc domain is a network made up of mobile/stationary vehicles furnished with OBUs and stationary RSUs that are installed along the road. OBUs of different vehicles form VANET, where an OBU is equipped with communication devices, including at least one short-range wireless communication device dedicated for road safety and other safety applications. The RSUs allow the OBUs to access the infrastructure and as a result get connected to the network resources including the internet [28].

Apart from the application unit, the OBU is also equipped with the following hardware [28] [29]:

- A Central Processing Unit (CPU) that implements the applications and the communication protocols. CPU has a set of on-board computing devices allowing the vehicle to perform applications.
- A wireless transceiver that transmits and receives the data to and from the neighbouring vehicles and the RSUs.
- A location sensor in the form of GPS receiver that captures the relatively accurate positioning and the time synchronisation information.
- Various sensors to measure different parameters that need to be measured and eventually transmitted. The information could be reporting crashes, engine statistics, weather conditions etc.
- An input-output GUI that facilitates human interaction with the system.
- A set of Communication devices that allow the OBU to communicate with other vehicles and RSUs [28].
- The OBUs might also include memory to store and retrieve some frequently accessed piece of information, some additional UIs, equipment and network interfaces, and security devices. The memory and the UIs are further classified as follows:

- MEMORY:
  - Storage Pages: The most basic type of memory in the OBU is called the 'application-controlled' storage page. This type of page is used by the AU to store and retrieve frequently accessed information such as the driver's data. Typically, the implementer of an AU asks that a large number of OBUs be manufactured within which their specific page is reserved. This may be done by the manufacturer or at a facility operated by the implementer of the AU. Thereafter, the AU uses the services of the storage page of the OBU memory to store and retrieve data. The data is stored in a format characteristic to the AU and is not changed by the CPU of the OBU in any way.
  - Memory-Mapped Pages: If the OBU has a combination of UI devices, then those UIs are controlled by writing data to, or reading from, special memory pages denoted as 'memory-mapped pages'. These memory-mapped pages act as buffers to the specific UI associated with each such page.
  - Transfer Pages: Pages designated as transfer pages provide a means of interfacing between the OBU equipment and networks. These pages are later associated with a specific external RSU through the wireless DSRC interface.
- User Interfaces (UIs) Visual Displays: The OBU may offer visual displays such as coloured light emitting diodes (LEDs). There may be up to three different colours with red, green, and yellow as the standard colours or their closest equivalent depending on the display technology. Buzzers (Audible Alert): A buzzer may also feature on the OBU that alerts the user with an audible signal. The alert may be about a massage received from the neighbouring vehicle or

from a RSU.

- Enunciators: The OBU may provide enunciators which are devices capable of replicating speech or music. An enunciator is different from a buzzer in that it can be used to hand messages that have been written to the corresponding memory-mapped page whereas a buzzer only sounds 'on' and 'off' fixed tones.
- Character Readout: The OBU may feature character readout. The character readout is used to display text messages that have been written to the corresponding memory-mapped page.
- Keypad: The OBU may have a keypad. The keystrokes typed from the keypad are entered as a message into the corresponding memory-mapped page. Authorised applications can then access the entered data and read the page [29] [30].

A brief diagram of a vehicle with OBU and the components of the OBU is shown in Figure2.2:

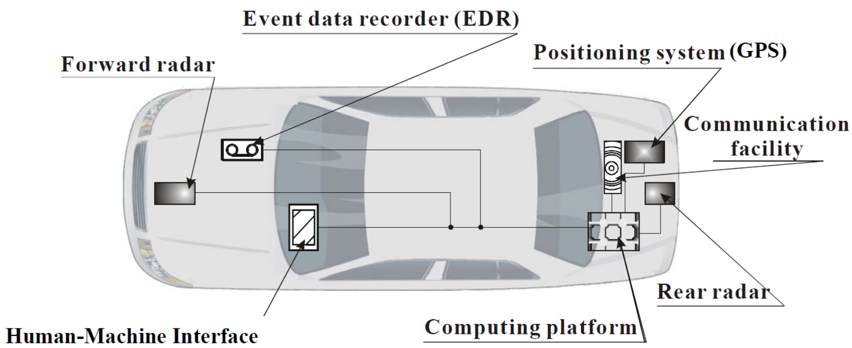


Figure 2.2: On-Board Unit components [31]

2.3.2.2 Road side unit

The RSU is a wireless access point which is located in urban areas and motorways to support V2I communication. This RSU based technology enables the use of a free spectrum with a high-speed connectivity (from 100 Mbps and up to 10 Gbps) within the RSU transmission range [32]. The RSU processes or forwards the information received from the vehicle OBU. As the RSUs are far away from each other in this case, the information is gathered through a non-continuous connectivity which makes it inappropriate for an interactive application such as safety applications.

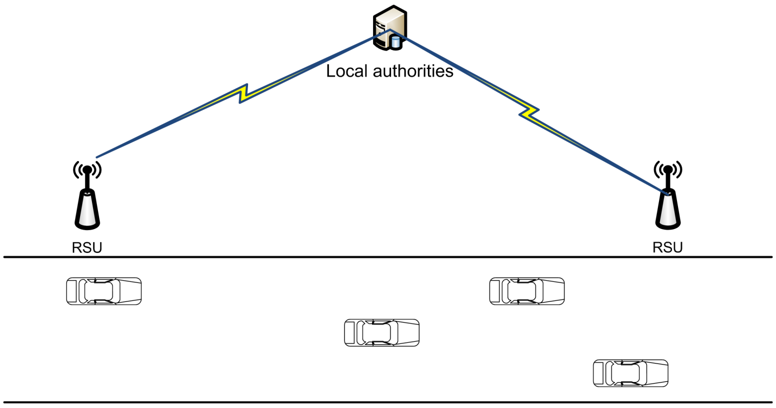


Figure 2.3: A fixed road side unit (InfoStation)

2.3.3 VANET Connectivity

Typically, VANET communicates over DSRC. The US Federal Communication Commission (FCC) in 1999 allocated 75 MHz of DSRC spectrum at 5.9 GHz for exclusive 'Vehicle to Vehicle' (V2V) and 'Vehicle to Infrastructure' (V2I) communications. Improving traffic flow and consequently saving lives appears to be the primary purpose of this spectrum. The spectrum also allows private services that help lower costs and promotes the development and adoption of DSRC [33].

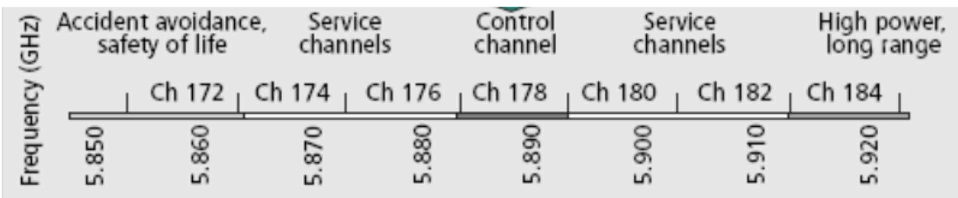


Figure 2.4: DSRC channels arrangement [34]

As shown in Figure 2.4 the DSRC spectrum comprises of seven 10 MHz channels. The control channel is set at Channel 178, and is limited to safety communications. The two channels at the spectrum edges are reserved for future advanced avoidance applications and high-powered public safety usage. The remaining are service channels available for safety and comfort uses [35].

The DSRC in Europe is often applied to a narrower set of applications, including electronic tolling. The EU gave a judgement in August 2008 to allocate 5.875 to 5.905 GHz of spectrum to ITS [36].

		Wireless Technology												
		5.9 GHz DSRC	2.5-3G PCS and Digital Cellular	Bluetooth	Digital Television(DTV)	High Altitude Platforms	IEEE 802.11 Wireless LAN	Nationwide Differential Global Positioning System	Radar	Remote Keyless Entry (RKE)	Satellite Digital Audio Radio Systems (SDARS)	Terrestrial Digital Radio	Two-Way Satellite	Ultrawideband (UWB)
Capabilities	Range	1000 m	~4-6 km	10 m	~40 km	120 km	1000 m	300-400 km	2 km	30 m	US 48 States	30-50 km	N/A	15-30 m
	One-Way To Vehicle	X			X	?		X	X	X	X	X		?
	One-Way From Vehicle	X				?			X					?
	Two-Way	X				?							X	?
	Pont-To-Point	X	X	X		?	X			X			X	?
	Point-To-Multipoint	X	X	X	X	?	x	X	X		X	X		?
	Latency	200 μsec	1.5-3.5 sec	3-4 Sec	10-30 sec	?	3-5 sec	N/A	N/A	N/A	10-20 sec	10-20 sec	60+sec	?

Table 2.1: Comparing wireless technology by NHTSA [23]

The DSRC radio technology is identical to IEEE 802.11a, but modified to operate in the low-overhead environment of the DSRC spectrum, and is being standardised as IEEE 802.11p. The IEEE 1609 working group is standardising the overall DSRC communication stack between the link layer and applications. The following standards have been proposed for VANETs by the IEEE [23]:

- *IEEE P1609.1*: standard for the Wireless Access for Vehicular Environments (WAVE) resource manager. It defines services, interfaces and message formats.

- *IEEE P1609.2*: standard for vehicular network security, including message formatting, processing, and exchange.
- *IEEE P1609.3*: a standard that defines routing and transport services, and is an alternative to IPv6.
- *IEEE P1609.4*: provides multiple channel specifications in the DSRC standard.

For its MAC layer protocol, the WAVE stack uses a modified version of IEEE 802.11a known as IEEE 802.11p and uses Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) as the basic medium access scheme. Through the orthogonal Frequency Division Multiplexing (FDM) system, it provides a 1000 metre communication range and handles high absolute and relative velocities, fast multi-path fading and different scenarios [23].

2.3.4 Communication Deployment

Vehicular communication systems are an emerging type of network communication technology, in which the basic communication nodes are the 'vehicle' and the RSU. These nodes are provided with an in-built capability to use wireless coverage to send essential information such as safety warning and traffic updates to neighbouring or distant nodes. The structure of communication in VANET is categorised under the three main classes as follows:

1. *In-Vehicle Communication (InVC)*: takes place between the various units inside the vehicle such as between the OBU and application unit (AU) which are quite common in modern cars. Generally, the InVC has two applications: the first is the in-vehicle network of sensors, actuators and controllers that collect and send the information to the OBU; and the second is high rate multimedia

communication for comfort applications for example, passenger entertainment, parking spot location and reservation, online video streaming for non-drivers, to name but a few [37].

Seeing that the number of communicating entities is not expected not to change over the vehicle lifetime in most of the situations, the topology of in-vehicle communication networks is a stable one; obviously, it defines a restricted group of possible communication partners and relies on wire-line communication.

Controller networks and security applications have particularly tight requirements on delay and integrity, whereas in the case of comfort applications, consequences of violations of the maximum allowed in delay or data corruption are less serious, but higher data rates are required [38]. The necessity of communication system standardisation for in-vehicle network comes from the constant increase in the number of integrated electronic components, credited to the advances in CMOS and VLSI technology [39].

2. *Inter-Vehicular Communications (IVC)*: In recent years, many researchers have turned their attention to IVC which is known as V2V as well, particularly in the USA, EU and Japan, owing to its ability to expand the driver‘s prospects which leads, as a result, to enhancing the safety issues. The safety issues may include collision warning and avoidance, safety of cyclists and motorists relating to road traffic and thus increasing the efficiency of system safety by providing virtual human interaction, i.e. interaction among the nodes (vehicles) itself [40].

   In this type of communication, vehicles are managed in a decentralised way, by allowing the vehicle to initiate direct communication with other vehicles without any support from the infrastructure, IVC communication can act as a crucial building block of an ITS. It is considered to be a more realistic solution if infrastructures are unavailable [31] and a much more economic solution if the high cost of installing and deploying any type of these infrastructures is taken into account.

The 0.3 GHz – 300GHz band of the electromagnetic spectrum known as the Microwave region is considered to be the most important stream used in IVC; in particular, the DSRC which was allocated by the FCC (Federal Communication Commission) in the U.S. utilises spectrum over 75 MHz in the 5.9 GHz band, while in the EU and Japan the 5.8 GHz band is used [41].

3. *Vehicle-to-Infrastructure Communication (VIC)*: Vehicle-to-Infrastructure Communication or V2I indicates the message transfer between the RSU and the OBU, and is also known as Road-to-Vehicle Communication (RVC). This cellular WiMAX based topology is considered to be a relatively expensive alternative, owing to its requirement of installation of a large number of RSUs and Base Stations (BS). These RSUs and BSs are responsible for coordinating a number of connections between subscribers in cellular networks to ensure a complementary cover to all the roads [5].

   The base station here plays the main role in controlling the negotiation process and setting up the connection between the infrastructures and mobile vehicle. The RSUs employ the WAVE standard [42] that uses DSRC to communicate with vehicles. Moreover, V2I communication is based on cellular standards of GSM, UMTS and WiMAX to provide internet access and infotainment applications.

   The term 'infotainment' refers to information-based media content or programming that also includes entertainment content in an effort to enhance the popularity and to maintain user interest. Some of the representative infotainment applications include navigator, audio/video systems, internet applications, etc. For immediate response from infotainment devices on user control, the Media Oriented System Transport (MOST) specifications make it necessary for the control messages to be processed in real-time to fully utilize the network bandwidth and to maintain the Quality of Service (QoS) [43].

As illustrated in Figure 2.5, V2I provides a facility that enables vehicles to initiate a connection with other fixed units that are distributed along the road. A vehicle can do this by disseminating useful information describing the vehicle's current situation, which can be used to support the decision that can be taken by other vehicles in terms of movement action, direction and speed. Alternatively a vehicle can also do this by receiving similar information from other vehicles to avoid taking certain action or to take certain necessary action [44] [45].

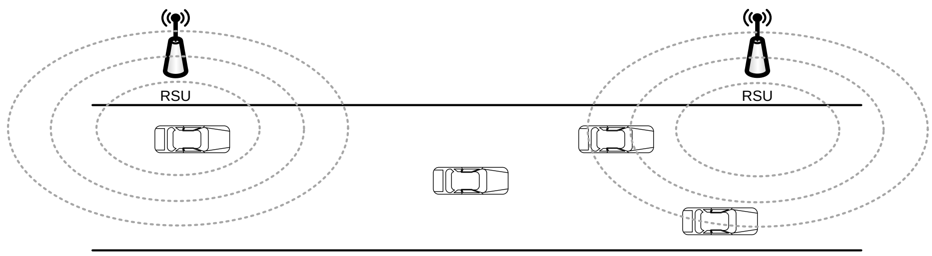


Figure 2.5: V2I Communication in VANET

2.3.5 VANET Applications

There are two categories of VANET applications: Safety applications and Comfort applications. The functionality of safety applications includes monitoring surrounding roads, approaching vehicles, road surfaces and road curves. These applications exchange messages and work together to assist other vehicles. They may also automate certain driving features such as emergency braking to help avoid accidents [46].

Comfort applications are primarily traffic management applications and as such, enhance the efficiency of traffic by increasing driver convenience, as well as providing entertainment and services such as web access, streaming audio, streaming video and gaming [46].

2.3.5.1 Safety applications

Safety applications include slow/stop vehicle advisor (SVA) applications that work by broadcasting warning messages to neighbouring vehicles when his/her vehicle is driving slowly or has stopped. Other possible applications are the emergency electronic brake light (EEBL)application or the post-crash notification (PCN) application which allows a vehicle involved in an accident to broadcast warning messages regarding its position to trailing vehicles that can then make relevant decisions on time, and to the highway patrol for towing [47] [48].

The cooperative collision warning application alerts two vehicles that may collide to make the appropriate adjustments. Road feature notifications may also alert drivers in advance about road curves and sudden downhills [46] [23].

VANET safety applications can be grouped into five major categories [49]:

- *Intersection Collision Avoidance:* V2I communication forms the basis for intersection collision avoidance systems. This infrastructure includes sensors near intersections to collect data on nearby vehicle movements. The data collected is processed and analysed to determine unsafe situations that may result in accidents. A warning message is sent to vehicles that have the potential to be involved in an accident. As this mode of V2I communication has been successfully implemented, the same methodology can be used for comfort applications, such as for V2I communication to locate parking [49].

- *Public Safety:* The focus of most public safety applications is to minimise travel time for emergency vehicles. Such applications are designed to help these vehicles provide efficient services and to help the drivers when required. Public safety applications also assist drivers during accidents and help to avoid subsequent accidents [49].

- *Sign Extension:* Recently, drivers have begun to use new technology devices such as cell phones and Personal Digital Assistants (PDA) while on the road. Such use can be distracting and result in accidents from reckless driving. Sign extension applications focus on maintaining a driver's alertness regarding road signs and structures that may cause accidents [49].

- *Vehicle Diagnostics and Maintenance:* Vehicle diagnostic and maintenance applications alert vehicle owners when it is time to have their vehicles checked and about safety defects. Such defects may also be related to tires and breaks and to other parts of a vehicle [49].

- *Information from other Vehicles:* Short-range communication facilitates the transfer of information from other vehicles' applications. In-vehicle applications use such information (for example, position heading) to complete their functions [49].

2.3.5.2 Comfort applications

The comfort applications (or the non-safety application) provide comfort and more traffic efficiency for drivers and passengers. Such applications may include providing current traffic or weather information, interactive communication, and nearest point-of-interest (POI) localisation.

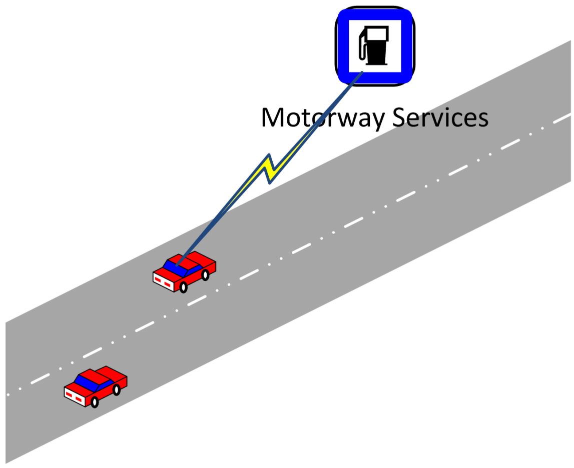


Figure 2.6: Motorway services providing vehicles with the latest prices information

A variety of applications developed to run on top of a TCP/IP stack may be created, such as online games and instant messaging. Wireless advertising also has potential, and the technology can allow enterprises (such as shopping malls, fast food establishments, gas stations, and hotels) to set up stationary gateways that transmit marketing information to vehicles passing by. Crucially, the comfort applications should not interfere with the operation of safety applications. Prioritising traffic and using separate physical channels is viable here [50].

It is however apparent that these detached approaches in the design of parking systems will struggle to give the necessary flexibility and results with wider acceptability. Instead of using a few approaches in isolation, a more innovative approach needs to be employed to design a mechanism to rapidly locate and reserve a vacant parking space.

2.3.6 Wireless Access Technology

Several wireless access standards exist that can be used as a base for connectivity through VANET. Using the available communication media, a set of air interface protocols and parameters can be specified for high-speed vehicular communications.

These core technologies include:

- Cellular Technology (2/2.5/3G): The advantages offered by the 2/2.5G technology are coverage and reliability, with 3G slowly taking over as the preeminent technology and providing improved bandwidth and capacity. There are several telematic and fleet management projects using cellular technology (SMS reports). However, high costs, limited bandwidth and latency render this technology unfeasible as a primary means of communication [51].

- IEEE 802.11p Based Technology: IEEE has time and again come up with new variations of 802.11 standards to support communication between the vehicles or between the vehicles and the RSU. The IEEE 802.11p is the amendment to the IEEE 802.11 standard to add WAVE. The 802.11p standard defines enhancements to 802.11 required to support ITS applications. This includes data exchange between high-speed vehicles and between the vehicles and the roadside infrastructure in the licensed ITS band of 5.9 GHz (5.85-5.925 GHz).

  IEEE 1609 is a higher layer standard on which IEEE 802.11p is based. This technology can satisfactorily work with speeds of up to 200 km/h and handle communication ranges as high as 1,000 metres. To promote this technology, the initiative has already been taken by the auto industry in Europe (Car2Car Communication Consortium - Car2Car CC) and in the US (Vehicle Safety Communications Consortium VSCC, Vehicle Infrastructure Integration VII). The structure deployment costs are expected to be low given the large production volumes [51].

- Combined Wireless Access: The challenging task of combining all the existing wireless communication standards to provide an exhaustive and high quality VANET service has been done by ISO TC 204 WG16, known as the CALM M5 (Continuous Air Interface for Long and Medium Range). CALM M5 has been one of the most significant efforts to date to combine wireless access technologies and it builds on the IEEE 802.11p by incorporating a set of additional interface protocols.

CALM M5 supports several standards, including the cellular systems GSM, HSCSD, GPRS (2/2.5G) and UMTS (3G) and infrared communication and wireless systems in the 60 GHz band. Integrating these systems results in greater flexibility and redundancy which improves the application performance. In addition to addressing inter-operability, CALM also addresses the issues pertaining to protocol standardisation, the network layer, and management services [51].

2.3.7 Challenges in Use of VANET

Deploying a vehicular networking system requires that several issues have to be resolved that require expertise from a variety of areas, such as applications critical to economical concerns. VANET has traffic, safety and user application based challenges which have some specific design requirements [51]. Apart from these general challenges, a few more specific challenges are present in implementing VANET:

- Due to mobility of vehicles, the network topology changes rapidly resulting in small effective network diameter.
- The number of vehicles passing through a particular junction is different every day, thus making the network density highly dynamic.
- There is a possibility that the driver might not adjust his behaviour, i.e. not react to the received data and thus render the VANET service futile [52].

Therefore, the following core research challenges need to be addressed when deploying such a network:

2.3.7.1 Spectrum issues

The Federal Communication Commission (FCC) of the US and its UK counterpart Ofcom together with National Telecommunications and Information Administration (NTIA) have reported that most of the spectrum is idle most of the time. The licensing and allocation of spectrum is a sensitive issue that depends on a number of crucial factors like public safety, affordability and maximum bandwidth usage, energy conservation issues, and spectrum reservation for research and development purposes. With broadband, internet and digital TV becoming popular, recycling of the TV spectrum and the necessity to clear the TV white spaces has increased the need of innovative ways to handle the spectrum matters [53].

Amidst all these controversial issues surrounding the spectrum allocation and licensing policies, FCC has allocated 75 MHz of spectrum at 5.9 GHz (from 5.850 to 5.925 GHz) to V2V and V2I communications. The VSCC and VII consortia have agreed that a derivative of IEEE 802.11 would be the optimal technology for communication systems at this spectrum [51].

In Europe, a continuous 75 MHz spectrum in DSRC band is not available. Therefore, Car2Car CC proposed an alternative approach that includes allocating two 10 MHz for use primarily for safety critical applications in the 5.9 GHz range (5.875 to 5.925 GHz). The comfort and commercial applications would also use an additional spectrum either in the 5 GHz Remote Local Area Network (RLAN) band or in the 5.8 GHz IRM band for non-safety critical and commercial applications [53].

This allocation in Europe allows for worldwide harmonisation as this band is used as a control channel in the US. Although the 5.9 GHz band is currently allocated to military radar systems and fixed satellite services, the CEPT/ECC (European Conference of Postal and Telecommunications Administrations/Electronic Communications Committee), the Short Range Device Maintenance Group (SRD/MG) has recommended allocating the 10 MHz control channel at 5.885 to 5.895 GHz in line with the US approach. The second 10 MHz channel would be allocated in the upper part of the Industrial, Scientific and Medical radio bands (ISM) band at 5.865 to 5.875 GHz, taking into account radio-location services below 5.85 GHz [51].

2.3.7.2 Broadcasting and message dissemination

As future applications are expected to consume a significant bandwidth to broadcast information, several broadcasting techniques need to be considered. Such techniques may include FM radio (used for RDS/TMC Radio Data System / Traffic Message Channel) in the narrow band, as well as wider bandwidth digital services such as Digital Audio Broadcasting (DAB), Digital Video Broadcasting (DVB) and Digital Data Broadcasting (DDB). An alternative solution is the satellite broadcasting given its real-time, high traffic information services [54].

Its low cost and ability to handle large volumes of data make broadcasting appear an attractive solution. Currently, a few services based on DAB broadcast and Transport Protocol Experts Group (TPEG) protocol offer real-time traffic information [35].

Location-aware broadcasting limits the range of broadcasting to just the area of interest. This reduces overheads by avoiding the broadcast storm problem. Clustering, another method of optimising message dissemination, allows neighbour nodes to form manageable units, or clusters, to limit broadcasting range. One clustering method, Local Peer Groups (LPGs), allows nodes to form either static or dynamic clusters [51].

2.3.7.3 Routing issues

Although MANET routing protocols have received significant attention in the past few years, certain network characteristics of vehicle-related networking make MANET protocols unsuitable for use in vehicular networks [55]. MANETs have been unsuitable for vehicular applications primarily as a result of the MANET routing assumption that an intermediate node is always required between a source and a destination node, implying that an end-to-end connection can always be established [56].

However, VANETs' frequent network partitioning requires the 'carry and forward' concept where a node carries a packet until it can be forwarded to another node closer to its destination when no direct route exists. This concept works with one of three main routing algorithm categories used by VANET: the opportunistic forwarding; trajectory based forwarding; and geographic forwarding [57].

In the context of VANET, geographic forwarding and trajectory forwarding work similarly since vehicular traffic follows the layout of roads. With GFG/GPSR (geographic forwarding), a node's geographic location facilitates the forwarding of packets to their destination, and a simple Cartesian distance is used. Although providing good scalability, this method has problems with dead ends and voids, even with a path to a destination. Perimeter routing serves partially to solve the problem [57] [58].

With trajectory routing, road infrastructure is used as an overlay directed graph, intersections are used as graph nodes, and roads are graph edges. Here, messages follow predefined trajectories and distance is the graph distance, resulting in the view that trajectory routing may be the most natural message forwarding algorithm used by VANETs [51].

2.3.7.4 Power management

Power management in VANET is concerned with transmission power as opposed to efficiency. Too high power levels and the resulting interference can disrupt transmission at far away nodes. With respect to routing, a consideration is how transmission should be adjusted to maximise the overall throughput and minimise interference [51].

2.3.7.5 Security and privacy

The area of security must be assessed and addressed carefully when designing a vehicular communication system, as potential threats (fake messages causing disruptions or danger, compromising of drivers' private information) can disrupt operations. Vehicles must be able to trust the messages they receive; the system should be resilient and efficient, and messages should be authenticated in real time [59] [8].

User privacy is critical to any application, and communication network also must not reveal the identity of vehicles nor allow for tracking of vehicles by non-trusted parties. To avoid future lawsuits, privacy concerns must be addressed during the initial design stage of a network. In the networking world, each node carries a unique, permanent Media Access Control (MAC) address, allowing it to be traced. To avoid such a scenario for vehicles, IEEE 802.11p provides for dynamically assigned MAC addresses and a mechanism for duplicate MAC address discovery [51].

2.4 Overview of the InfoStation Concept(IS)

In this section, a new wireless system concept called InfoStation is analysed that can provide isolated pockets of high bandwidth connectivity for future data and messaging services [60]. The intermittent connectivity available to mobile terminals using a network of ISs raises new issues in protocol design on several levels [41].

2.4.1 Background

The IS, an innovative concept for a system that supports 'many time, many where' offers low-cost and high bandwidth wireless communication data services [61] [1]. A brief investigation has revealed that IS was first introduced in the late 1990's when the high commercially available data rate provided by IEEE 802.11b was 11 Mbps. Since that time, IEEE 802.11g, IEEE 802.11n, and IEEE 802.15.3a (UWB, ultra-wide band) have been developed until recently, and offer much higher data rates typically in the range of Gbps [46]. At such high speeds, IS related applications for information transfer can be supported even for time critical applications. Such applications assume that a mobile terminal randomly crosses an ISs' coverage area, and during the crossing the IS downloads the relevant and desired data [46].

Cellular systems today bring with them the promise of 'anytime, anywhere' communication. Voice messaging, email and fax are alternatives that return some control of time and place to the user, which may account for their dramatic growth. An array of isolated wireless ports called as ISs have been proposed [50] [34] for providing convenient and frequent access to high bit-rate connections. Individual ISs on highways, streets and in airports may function independently, providing specialised gateways to the Internet and accessing remote servers for information and messages

[46].

The purpose of introducing IS in this thesis is because of the fact that the wireless coverage provided by IS and accessed by the vehicle OBU can be used to make vehicles as networked devices. These smart vehicles should then be capable of carrying out wireless communication with the infrastructure and with other networked vehicles on the road [41].

2.4.2 Establishing the IS Network

From the IS based m-learning system proposed by [62], it is evident that ISs may be implemented easily and cost-effectively in highly populated, traffic stricken areas such as coffee shops, airports and hotels. A radio transceiver provides low-power, high-data-rate Internet access from an IS to mobile devices in close proximity to the station [63] [64]. Thus, in the future, an IS may even be located in the centre of a city, allowing users to download data while moving from one side of a city street to the other [65] [66].

However, the design of an IS to support the increasing number of mobile phone and smart phone users raises several key issues, including radio design, multiple access technologies, layout and coverage, and guaranteeing achievable payloads [2]. A solution to tackle this problem would be to install IS more frequently, at smaller distances from each other so as to provide a strong and unbroken coverage[67].

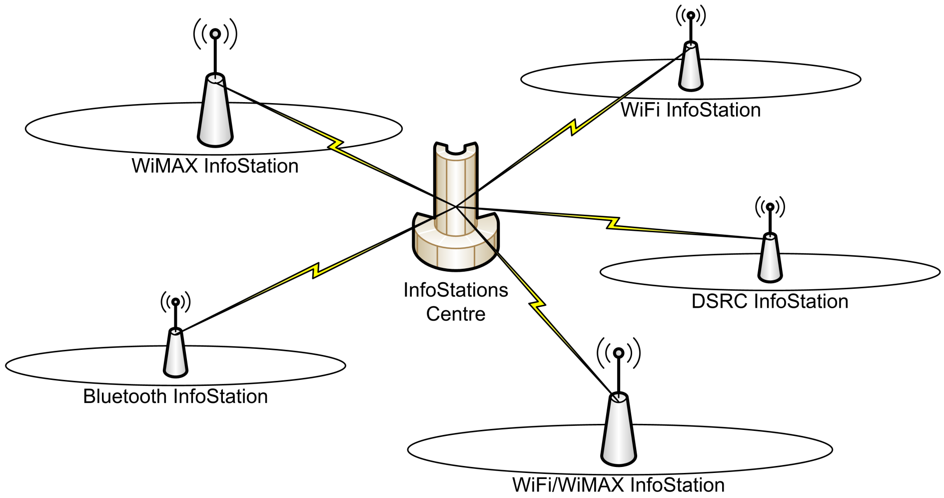


Figure 2.7: An example of the InfoStation network

Thus, a few advantages offered by deploying IS in a vehicular network can be summarised as follows:

- IS provides a wireless coverage to vehicles thus making vehicles networked and communication-capable devices.
- IS installed in a particular parking zone can enable parking space information dissemination to vehicles, thus making interaction between vehicle and parking possible.
- IS, together with safe wireless DSRC communication can enable secure electronic payment for parking, thereby increasing the comfort level for drivers.

2.5 Overview of the Context-Aware Systems

2.5.1 Background

The term 'Context' is widely used in Computer Science and Software Engineering and other related fields with different meanings. The absolute meaning of the term 'context' depends on the work area in which it is defined.

For instance, in Artificial Intelligence, the term context is defined as something that can have an effect on the computations being carried out and includes the factors both internal and external to the computation. Thus, context refers to the physical and social situation in which computational devices are embedded [68].

In Natural Language Processing (NLP), context is conceived as all the knowledge that gathers around a specific statement. In the field of NLP, it is of primary importance not to consider the things that are 'out of context'. Statements that may be true in one context might not be so in another and might be absolutely false in a third context. Thus, context here is dependent on the meaning of one sentence with due regards to the meaning of other related sentences [69].

In Operating Systems,context defines the smallest amount of data required to be used by an operating system's task, and which needs to be saved in order to allow the task to be interrupted and to be resumed later from the point where it was interrupted. In Ambient Intelligence (AmI), the notion of 'context' is perceived slightly differently where there isn't a generally accepted definition. Here, 'context' is characterised by the region of use, the assemblage of entities nearby such as people, environmental conditions, and accessible devices [70].

Thus, context-aware systems can be defined as those that have the ability to adapt themselves to these surrounding aspects. The concept of context-awareness is described as the 'ability of the computer to sense and act upon information about its environment, such as location, time, temperature, or user identity.' The most commonly used definition of context in AmI systems is offered by Dey who defines context as 'any information that can be used to characterise the situation of an entity' where an entity may be a person, a place, or an object relevant to the current interaction between a user and an application [70].

Researchers have also classified context in two separate classes, namely personal context and environmental context. Environmental Context might include factors such as the time of day, the opening times of public places or the most recent weather forecast. Personal Context, on the other hand, refers primarily to user profiles such as user's interest, experience, attitudes, or beliefs.

Another basis of classifying contexts has been in the form of user's context and system's context. The user's context explains how observations are made and to interpret the observations. The system's context provides a means to compose the federation of components that observe the user's context.

Interestingly, various researchers have accepted the formal definition of context as 'the necessary and sufficient information about a location and its environmental attributes.' For example, this might include the examples described in [71] such as a weather forecast, opening times of public places, a time of a day or other subject specific factors for instance noise level, temperature, light intensity and movement and the people, objects, devices and software agents that it contains.

Context might also include the system capabilities to process a given signal and produce an output that is different from the input, the activities and tasks in which people and computing entities are engaged, services sought and services on offer, their situational roles together with their respective beliefs and aimed intentions.

2.5.2 History of Context-Aware Systems

Schilit and Theimer (1994) were the first to introduce the term context-aware in 1994 where context was defined as location, identities of nearby people, objects, and changes in these objects over time. Dey and Abowd (2000) defined context as 'any information that can be used to characterise the situation of entities that are considered relevant to the interaction between a user and an application, including the user and the application themselves.' This has become the most widely accepted definition of context. [72] [73].

2.5.3 Specific Terminology

A few common terms used for embedded computer systems are Ubiquitous Computing / Ambient Intelligence / Pervasive Computing. Although these terms are often used interchangeably, the primary differences among these terms depend on the context in which the term is used.

The term ubiquitous computing is generally presented in an academic sense as a 'manageable and human-centric technology', currently in the research phase and will be realised in the years to come [74].

The term 'pervasive computing' has been coined by researchers from IBM and it has a slightly similar implication. The basic idea is to move beyond the personal computer and to get everyday devices being embedded with technology and connectivity and also get computing devices to become progressively smaller and increasingly powerful. Quite similar to ubiquitous computing, pervasive computing is the result of computer and wireless technology progressing at exponential speeds; a trend moving towards all the products having hardware and software.

Pervasive computing goes beyond the realm of laptops, smart phones and PCs, and proposes that almost any device, ranging from clothing, coffee mugs, household tools and appliances, cars and motorbikes, homes and most-strikingly the human body can be embedded with chips to connect to an infinite network of other devices. Pervasive computing aims to combine the technologies such as wireless computing, artificial intelligence, voice recognition, and imaging systems to create an environment that has an embedded and unobtrusive, always-available connectivity [75].

A similar term coined in Europe for research into the topic is 'ambient intelligence' which has a lot in common with ubiquitous computing. A large number of technical events, conferences and research projects organised in Europe under the term 'ambient intelligence' have topics in congruence with the ones that are equally accepted in the ubiquitous computing field, albeit with a few subtle differences between these terms based on their etymological meanings. It can be stated in general that the basic idea behind these terms is the same and they can, therefore, be used equally in this thesis. And for the sake of reducing ambiguity, the term 'ambient intelligence' will be used throughout with the general meaning, unless otherwise stated [75].

2.5.4 The Constituent Parts of a Context Aware System

The three essential functionalities of a context aware system are: sensing, thinking and acting. Systems can be skewed in sophistication towards any of these functionalities, so where one system maximises performance by utilising a lot of deliberation and little sensing, another may use complex sensors but perform little reasoning before acting. What's more, these functionalities can be set within a centralised or distributed architecture and over one or more physical devices.

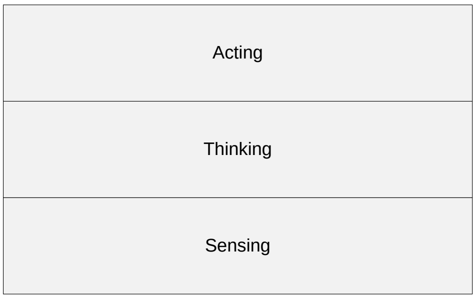


Figure 2.8: The Constituent Parts of a Context Aware System

2.5.5 Acquiring Context Information using Sensors

Acquiring context information is undoubtedly the foremost prerequisite for any system to be context-aware. In the simplest terms, context acquisition can be thought of as a process where the real world situations are captured, the most significant features of those captured are formally assessed and an abstract narration is constituted. The universally accepted and prevalent methods to acquire context information and make systems context-aware include location tracking, computer vision and sensor systems along with Bayesian and predictive modelling of users and their behaviour [76].

A mechanism must be in place for applications to be able to use context information, and to deliver such information to the application. In the past, researchers have devised various types of sensors for different purposes as listed below [77] [58]:

- Light Sensors: A single optical sensor that provides information on light density, reflection, wavelength and type (natural or artificial).
- Vision: A sensor that provides a wide range of visual information about the environment, an object's shape and recognition. Because they are easy to use and inexpensive, cameras are commonly used.

- Audio: A variety of methods at different processing costs can extract context information from audio data for analysis. Simple equipment, such as a microphone, can determine type of background noise, loudness and base frequency. With more advanced audio sensing and processing, such as identifying a speaker, more sophisticated context can be obtained. Using ultrasonic sensors provide an interesting aspect to audio context, such as using this equipment to augment human sensory capabilities [78].

- Motion Sensor: Mercury switches, angular sensors, and accelerometers are typical sensors used to detect motion. A moving car, for example, is detected by data provided by an accelerometer.

- Location Sensor: Pervasive computing uses parameters such as position, proximity of users, location, co-location, devices and environment as a measure of acquired context. GPS is often used outdoors for fine-grained location sensing and cellular network infrastructures. Location sensors are typically embedded in an in-door environment, such as using radio beacons to sense co-location.

- Bio-Sensors: These sensors can be embedded in the human body to measure pulse, skin resistance, and blood pressure, for use in sports and medical applications.

- Surround Sensors: Touch, temperature, air pressure, and other sensors may be used with more specialised applications, implying an unlimited use for context sensors [79].

The two classes of context sensors, namely location sensor and surround sensor can continue to support the wider range of applications in Intelligent Parking design. A context aware feature added to the vehicle can assist in the intelligent parking design by sensing the immediate surroundings and gathering relevant information.

This information could be the size of available vacant parking to check if the parking size is suitable enough to accommodate the vehicle or the distance between two vehicles parked adjacent to each other.

The location sensors can be of particular importance in transmitting the location information to the IS and aid in locating a vacant parking at a desired destination.

In brief, the advantages offered by context-aware vehicles in facilitating a smart parking system can be summarised as follows:

- Sense information about the surroundings and the location
- Share broader information with the infrastructure to facilitate a vacant parking location and to increase security and traceability
- Enable real-time information sharing among vehicles and infrastructure.

The main objective of locating parking can be achieved if the vehicle information and parking information can be matched and it can be decided whether the vehicle is suitable to be parked in a given parking space.

Sensing and sensor technologies are extensively used in fields such as robotics, industrial automation and production engineering [80]. In these environments, sensors are an essential source of surrounding information that assist in creating useful solutions. The eventual goal of acquiring this information through sensors is to have a representation of the world around which approximates the perception of the user.

Based on these investigations, various types of sensors are introduced and assessed for their utility for building context-aware systems with respect to the implied requirements. Therefore, a surround sensor mounted on the vehicle bumpers should be able to measure the distance between adjacent vehicles in a parking zone so that a minimum gap between two vehicles is easily maintained. The location sensors in the vehicle that use GPS to sense the current position of the vehicle and transmit that information to the IS are also necessary in making the vehicle context aware [81].As touched upon in the earlier section, context is solely based on location together with the concepts of 'cognition, perception, recognition and abstraction' [40] [82].

2.5.6 Challenges to Overcome In Designing Context-Aware Systems

Several examples, simulation and prototypes have resulted from the extensive research into context aware computing, demonstrating that context awareness can positively enhance the working and the suitability dimensions of systems and applications. In acquiring context information, a location is sensed and then combined with a few assumptions, a more general context aware system is made.

Though the concept of position and location provides an impressive and uncomplicated model for context-aware applications, yet in a number of cases awareness solely based on location might lack some fragments of information that can be essential for a system to make it strongly context aware [81] [56]. To acquire information beyond location would be to add another layer of complexity in sensor applications. The following issues cover the challenges that are central to the research in context-awareness:

2.5.6.1 Positioning sensors for intelligent parking systems

A variety of positioning sensors exist, depending on the specific need. These sensors are used to detect the activities and locations of people in an environment and are worn by humans or placed in a room or vehicle. Sensors in vehicles help to automate certain functionality, such as detecting rain and turning on windscreen wipers, detecting darkness and turning on headlights, or detecting bad driving by others or being in too close proximity to another vehicle and generating warning signals. Wear and tear of car parts can be automatically detected using in-vehicle telematics, which diagnose and report such problems. Not all cars enjoy this functionality, and many such devices are still in the experimental stage. An increasing trend towards using such devices exists, fuelling the way for the smart vehicle [58].

2.5.6.2 How to make use of context

Once the context information has been made available in a system, the next question that arises is to identify the usefulness of the context. Another question that becomes imminent is to spot the range of applications that can be successfully enhanced with the acquired context information. Issues of reliability of the acquired information and then to resolve ambiguity, if any, need to be addressed. The context information acquired through surround sensors and location sensor efficiently need to be transmitted to IS and ISC through wireless DSRC, using the wireless coverage provided by the IS.

2.5.6.3 Associating context acquisition to context use

In any context-aware system, there is expected to be a somewhat close conjunction between context acquisition and context use. When a system becomes context-aware, its behaviour is pre-disposed to the general situation /context of use. The acquired information through sensors should be relevant to a specific application for which it is intended. For instance, the information acquired by sensors mounted on the vehicle should assist in either safe driving of the vehicle by avoiding collisions or should notify the driver if the particular parking spot is large enough to accommodate the vehicle.

2.6 Earlier Work Done in the Domain

Having defined the primitive concepts and technologies to be used as a basis for this research, the target is to develop a novel architecture for locating and reserving a parking space using intelligent mechanisms in order to find the most appropriate parking that best matches the driver's preferences and the vehicle's profile without distracting the driver. This approach exploits the knowledge and characteristics of context-aware systems, Vehicle ad-hoc networks and IS infrastructure.

The next theme to be addressed is the motivation that can be drawn from previous work along similar lines. A glance at reliable literature reveals that researchers in the field have taken into account the fact that vehicles now-a-days are increasingly equipped with modern communication accessories, capable of transmitting signals and information in a wide frequency range.

A deeper analysis of the literature available in the field of intelligent and smart parking systems has been presented in this section, with an aim of assessing the current state-of-the-art in smart parking system design. The mechanisms proposed and employed for 'locating and reserving a vacant parking space' has been the central point of focus.

2.6.1 Parking Lots and Vehicle Entry to the Parking

Much of the current, reputed and popular research reviewed in journals reveal that automatic parking systems allow customers to drive their cars into a parking bay and lock the car, at which point a computer-controlled mechanical system takes over.

When picking up a car, a customer enters a code and password and the mechanical system retrieves the car in minutes. Automatic parking systems come with a variety of vehicle detectors installed and are excellent in areas with limited space [20] [83].

All of these smart parking systems are available to garage operators who may use them in combination for optimal service for their customers. An example is the new Dallas/FortWorth International Airport (DFW) 29-gate terminal and adjoining eight-level parking garage which provides 8,100 parking spaces and utilises inductive loops to detect vehicles at all key points. The count data is relayed to a central server for processing, and uses the data to determine the occupancy of various areas and to display space-available information to drivers on dynamic message signs throughout the garage [20].

In [21], Gongjun et al. propose a wireless–based intelligent parking system especially applied to large parking lots. This parking system uses infra-red sensors to sense vehicles and parking belts to control vehicle entry and exit. Although this system provides a certain degree of control over vehicle entry and exit into a parking lot, thereby adding stability to parking use, it is worth trying to propose an architecture that eliminates the need of sensors on the parking zones. An alternative approach could be to use the surround sensors which are already mounted on the vehicle bumpers; this would inadvertently enhance the context aware characteristics of the system. The parking system, along with the vehicle's OBU will then use the collected information to calculate the size of the available space on the parking zone.

Verroios et al. in [31] have proposed a parking reservation system by deploying sensors on vehicles. Using VANET, their computational infrastructure informs the vehicle about an available vacant parking space. Whilst the vehicle is informed about the available parking, it is left to the vehicle to determine the time required to reach the parking because the parking can be occupied by another vehicle who reaches there earlier. A reasonable approach to tackle this issue could be to install IS and ISC to control the parking zones. With IS and ISC a payment based architecture can be introduced, where once a parking is booked by a vehicle, other vehicles cannot use that parking. This does not rule out the original VANET based system in its entirety, but only adds to it to avoid dissemination of the same parking information to multiple vehicles.

Jatuporn et al. in [20] propose a similar parking system which is restricted to parking lots that cut down on the deployment-extensive and costly infrastructure and where the available parking lots can still be located effectively. On identifying a vehicle at the entrance, the proposed wireless sensor mechanism locates a suitable parking space for a vehicle and also directs it to that location. The proposed approach declares wireless sensor heads to be responsible for indentifying vehicles from motorcycles and pedestrians. This piece of work can be extended further and the architecture can be applied to on-street parking zones which demand a slightly more complex management structure as compared to the parking lots.

A mechanism where a vehicle looking for a parking initiates a conversation with the infrastructure itself and also informs the infrastructure about the desired parking location, can be obtained by wireless DSRC communication between vehicle and infrastructure. More information about the non-employing of DSRC for this purpose can be obtained from Kenny [41] who has emphasised it's disadvantages on various applications which use DSRC widely. These applications mainly consist of collision prevention and collision avoidance. The architecture proposed by Kenny also includes OBU and IS. The infrastructure hierarchy can be extended to ISC and utilise the high capacity DSRC channel for comfort application of on-street parking reservation. In relation to this approach, the study conducted by Ghazy and Ozkul [33] integrates VANET with the RSU to use DSRC for resolving traffic congestion which is a safety application as much as it is a comfort application. This application eliminates many negative impacts when it comes to the economy and the environment especially during long distance commutes.

2.6.2 Sensor-Based Smart Parking Systems

Although a few significant achievements have been observed and reported in the above paragraphs with reference to the parking lot domain, introducing context aware vehicles would be a reasonable approach to simplify many complex architectures. Jie Sun et al. [78] have proposed sensor-based context aware smart cars imparting a 'complex reasoning' faculty to the cars. In the general architecture of these cars, a wide use of various sensors on the sides of the car has been proposed which sense data from the surroundings. A context-aware feature is proposed along the same lines by limiting the sensor to just two types: the surround sensor and the location sensor.

The argument here is that the data collected by these sensors will be sent to the IS through wireless DSRC, thus aiding the IS with better parking resource management. Further work has been established by yet another practical application by the name of 'E-Parking' that combines and streamlines reservation and payment systems for parking through advanced technologies. Such systems allow drivers to inquire about parking availability, reserve parking spaces at certain destinations, and make payments by cell phones, the Internet and DRSC. These systems require conventional detectors to detect approaching vehicles, and at the same time must be able to identify customers and their vehicles that have reservations and allow them to access their spaces. Such systems may also use confirmation codes that when a customer receives a message on the cell phone as part of the identification process at the parking lot [20].

In another study, a Parking Guidance Information System, or PGI has been proposed that provides information about parking space availability in major cities. Vehicle detectors installed at entrances, exits and individual parking spaces allow for the collection of data on available and occupied parking spaces. Frequently used detectors include loop detectors, ultrasonic, microwave, infrared and laser detectors, and machine vision detectors. These devices collect information such as 'empty' or 'full', number of spaces available, the exact location of available spaces and display them at designated locations to allow drivers to make the needed decisions [20]. These findings are subject to a few limitations as these studies did not evaluate the use of vehicle OBU to store the electronic payment details of the driver and communicate them to IS or ISC when needed through wireless DSRC which could eliminate the need of laser detectors, machine vision sensors, ultrasonic, microwave and infra-red sensors.

Among the first countries to implement smart parking systems were Germany and Japan. Most major cities in these countries have several smart parking facilities.

These facilities provide numerous benefits to both drivers and the parking operator, including:

- Allowing the customer to readily determine space availability before entering the garage and/or parking level
- Enabling the customer to plan for transit to public transportation with smart parking systems such as those employed at Park and Rides
- Allowing the parking operator to use the system data to develop or improve pricing strategies, predict future parking patterns and trends, and prevent vehicle thefts
- Enabling the parking operator to reduce staffing for traffic control within the facility
- Enabling significant reduction in traffic and resulting vehicle emissions by decreasing the time required for customers to locate open spaces.

One of the obvious improvements of smart payment systems has been to replace conventional parking meters by using advanced technologies to provide convenient and fast payment, improved fine collection rates, and reducing the number of assaults on parking officials [84]. Such systems use contact (credit cards) and contact-less (smart cards, RFID cards) methods, and mobile communication devices (mobile phone services). Payments made for specific parking spaces are used to determine vacant spaces [20]. The entire payment architecture could be efficiently integrated into a single framework, initiating the payment from the vehicle OBU with the debit/credit card details stored in the OBU. A vehicle looking for parking should be able to pay from the OBU to the ISC and reserve the parking for the desired duration. This would eliminate the need of much of the physical hardware needed as well as the human intervention that would be otherwise needed.

To summarise the investigation conducted so far, although there have been a few significant achievements in the parking lot domain, a much greater and deeper effort is required to ensure a fuller integration of various fields of research. The reviewed research work has individually addressed the various sub-themes of IS based parking systems, VANET based DSRC parking systems, and the sensor-based context aware parking system. The research to date has focused mainly on VANET and Vehicle to Infrastructure communication most probably because the VANET application has come a long way in designing smart payment systems as well.

2.7 Summary

In this chapter, a comprehensive review of the existing literature in the domain of ITS, smart parking design and related fields has been carried out with an aim of combining more and more disciplines of research together. Some related concepts such as VANET, IS and context-aware systems with their specific application to parking design have been the central theme of the chapter. The historical origin and general features of these three concepts have been presented with a view to making the reader aware of the state-of-the-art trends in application of these concepts in parking system design.

The highlight of the chapter was an insight into a number of significant pieces of research that have been carried out that use VANET, IS and context aware systems for vehicle identification, entry of a vehicle to a vacant parking spot, and secure payment for the parking. A detailed account and the key reasons for the widespread use of these concepts in parking design have been given.

The research has thrown up many questions that are worthy of being referred to for further research. It has been successfully argued that a single framework integrating the above mentioned three concepts should be tried to design a parking space location and reservation system. The significant points discovered were to use surround sensors and location sensors to make vehicles context aware, and transmit the acquired context information to IS and ISC by using VANET based wireless DSRC communication.

The chapter concludes with a discussion of the challenges that need to be faced in designing a novel IS based context aware on-street parking system. Many of these challenges have already been faced by other researchers and suitable solutions/alternatives has also been proposed by them. A few limitations that remain to be challenged form a significant part of this thesis.

Chapter 3

Preliminaries

Objectives:

- To introduce to the research behind the proposed idea
- To outline the research methodologies
- To highlight the key concepts of the research
- To enlist the Assumptions
- To present the mathematical notation in CCA

3.1 Research Introduction

The last few years have seen many researchers concentrating their efforts on enhancing the functional value of the VANET in order to utilise the network of vehicles available for various security and comfort based applications. The security and public safety based applications are mainly focussed on avoiding accidents and loss of life caused by vehicular driving [28].

With the steady increase of overheads, collision and contention, especially in high density networks, researchers have turned their attention to using VANET for security applications like collision detection and avoidance. Authors [31], [80], [85] and [3] have developed approaches for detecting and avoiding collisions between vehicles without the need for drivers to communicate independently. The proposed systems make use of sensors mounted on the sides of vehicles to keep track of safe distances between vehicles and alert the driver if the distance falls under the safe limit, thereby avoiding a possible collision. The safety applications have become possible largely due to the ability of VANETs to quickly and reliably disseminate data to a large number of vehicles.

The work in the thesis applies to on-street parking zones and the research focuses on designing mechanisms so that drivers will be enabled to locate and reserve a parking space even whilst they are on the move and have not yet arrived at the destination. The existing approaches need to be modified as they are expensive by making costly sensors to be mounted on the parking zones necessary.

A further cost is incurred when a mechanical vehicle-barrier is needed to control the entry and exit of vehicles in the parking. An element of uncertainty is also associated with these architectures as the parking availability information is disseminated to all the vehicles and the parking is reserved based on a first-come first-served basis. The architectures lack mechanisms to update other vehicles about the occupancy of the parking and thus leave a number of vehicles stranded and competing for one particular parking space.

Therefore, as an extension of VANET and wireless communication and as an improvement over the above mentioned technologies, the work carried out in this thesis uses the dual concepts of IS and context-aware systems in conjunction with the DSRC based wireless communication standard for locating and reserving a parking spot in the on-street parking zones. The information captured by the sensors mounted on the vehicles make the vehicles context-aware, and the IS installed at suitable distances provide a uniform wireless coverage to the parking zones.

This novel architecture aims to result in the decrease of the cost of existing intelligent parking system mechanisms, as the need for costly sensors on the parking zones, mechanical vehicle-barriers, payment machines and paper-based records for recording the parking status would be completely eliminated.

3.2 Research Methodology

The step-by-step research methodology used to achieve the predetermined objectives in this thesis are as follows:

1. Specifying the research problem which is accomplished by an extensive and concentrated literature survey

2. Simplifying the research by breaking the research problem into smaller problems that can be solved individually

3. Finding and developing techniques to solve these problems

4. Improving and expanding the techniques applied to the solutions

5. Evaluating and examining the research concept via the formalising and validation tools

6. Analysing the research strength and organising the documentation.

The investigation in this thesis has been established using a literature review covering VANET overview and smart-parking system in VANET. In the literature review, an overview was introduced in VANET to give a general observation of the VANET characteristics and applications.

It then discussed the smart-parking system in VANET in depth, as the main objective in this thesis is to design a new smart-parking system to improve the system's performance in terms of relieving traffic congestion and increasing the uptake of the system's usage.

3.3 The Key Concepts of the Research

The thesis has introduced the architecture of the IS based context-aware on-street parking system and two key concepts in order to manage the work flow of the architecture. The concepts include mechanisms that find a vacant parking space whenever a vehicle requests parking, and reserve a vacant parking space when the vehicle makes the necessary payment.

3.3.1 Locating and Reserving Parking using the On-street Parking System

This part refers to the parking space reservation, starting from the instance the driver sends a request for a parking space. The parking reservation scheme presented here is based on the concept of context-awareness. The design of vehicle parking system using the concept of context-awareness imparts an intelligent feature to the parking space's locating and booking system which has the potential to overcome the shortcomings of the existing parking system mechanisms.

The implementation of the system will use an OBU to communicate with the IS and a group of sensors that would enable the vehicle to gather information about the parking intelligently. The proposed system allows for communication between the vehicle and the parking taking place by the means of a set of ISs deployed alongside the parking zones. The context aware architecture is based on VANET that uses the DSRC standard to deliver packets from vehicle to IS and IS to vehicle.

It is presumed that each vehicle is equipped with physical surround sensing sensors. These sensors provide the OBU with the details of the location and distance of nearby vehicles in the parking zone. Thus, the OBU can help the IS to analyse the location and size of the free parking space and allocate it to a suitable vehicle.

Here, the system checks the vacancy of the parking space regardless of the location and size of the space of the parking zone. A mechanism for controlling the allocation and calculation of parking spaces to assist the IS to manage and utilise the free spaces in the parking zone using the vehicle sensors is necessary. Basically the task of the IS here is to check the validity of the vacant parking space to suit the size of the vehicle. Based on the vehicle information, the IS will start matching the vehicle size with the vacant spaces list and validate the parking spaces.

3.3.2 Evaluation of the On-street Parking System

The IS based context-aware on-street parking system is fully formalised in the CCA which is a process calculus for modelling and analysing the context-aware systems.

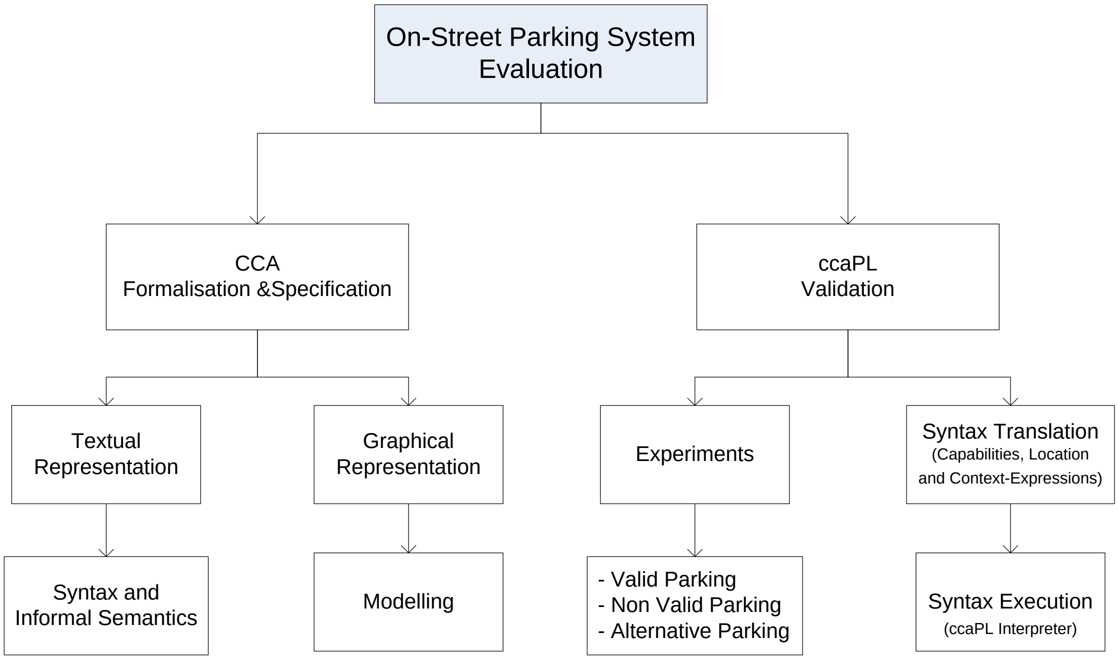


Figure 3.1: on-street parking system mechanism evaluation

The proposed parking reservation architecture has been formalised using the CCA notation in the form of Textual and Graphical representation. The execution environment of the CCA, namely the Calculus of Context-aware Ambients Programming Language (ccaPL) is used to prove and validate the parking reservation mechanism.

3.4 Assumptions

In order to achieve the remit of this thesis, the following assumptions have been taken into consideration:

- Vehicles are equipped with a number of sensors including GPS and surround sensors
- The vehicle profile, driver profile and the driver preferences are also a resource of contextual data
- Vehicles are equipped with OBU, preloaded with a digital map
- The parking zones are managed by the Local Traffic Authority
- Vehicles collect information about the parking environment and activity from nearby IS
- The system utilises the DSRC as the means of communication between the vehicles, IS and ISC.

3.5 Overview of the Concept of Ambient

In this chapter, we present the concept of the computational model which will act as a base for the forthcoming work. We begin with an introduction to the context-aware systems and then move on to the modelling methods of such systems. The fundamental concept in our computational model is that of the 'ambient' which has in recent years been an integral part of formal modelling. An ambient can be defined as a bounded region where some computation takes place.

An ambient is an entity used to describe any object or component (e.g. person, process, device or a location) in a system [86].By 'bounded region' we mean that there are two separate environments: one inside the boundary of the ambient, and the other outside the ambient boundary.

One of the most important properties of ambient is that an ambient can be nested within other ambient. For example, there can be an ambient called vehicle and another ambient called parking. But when that vehicle is parked in that particular parking then the ambient is said to be nested. Another property of ambient is mobility i.e. the ability to move from one location to another. Ambient moves as a whole; all the processes executed inside the ambient move collectively with the ambient. Recalling the Smart phone example, if the Smart phone is taken to a different location, all the files and applications contained in the Smart phone move accordingly and naturally.

An ambient is defined using the following universal structure:

- Each ambient has a name that differentiates the ambient from other ambient and helps in identifying the ambient. Identifying an ambient with a name organises and smoothens the moving in and moving out operations of the ambient, and facilitates the communication of an ambient with other ambient.
- Each ambient has a process which represents the behaviour of the ambient. These processes run directly within the ambient and define the capabilities of the ambient. For example, they can instruct the ambient to move.
- An ambient is allowed to have a collection of sub-ambient which are defined by its own name and can further have its own sub-ambient, thus allowing the set of ambient to be represented hierarchically.

An ambient can come in relation with other ambient and there are three possible relations among ambient: parent, child and siblings, as will be described in depth in the coming sections. Based on the relationships between the ambient, various terms are used to identify and define the environment of an ambient: parent ambient, child ambient and sibling ambient. In terms of context-awareness, an ambient is aware of its surrounding environments, i.e. an ambient has the ability to sense the context around it. For example, in the context aware on-street parking system, the ambient vehicle gains awareness of its surroundings with the help of the sensors mounted on the vehicle bumpers.

Thus, we are in a position to present a formal definition of an ambient: Ambient: is a bounded place where computations happen. It has a name, a boundary, and can have other ambient inside it. An ambient has the ability to move from one location to another as well as the ability to communicate with other ambient [87]. When an ambient has the knowledge of its surrounding environment, it is said to be context-aware.

3.6 The Calculus of Context-aware Ambients (CCA)

The dual concepts of mobility and context-awareness are fundamental to the development of pervasive computing systems that define the next-generation computers that are becoming increasingly smaller and subtly embedded into everyday life. As mentioned earlier, the concept of mobility implies that a component has the ability to move from one location to another and the concept of context-awareness states that the computations (processes) taking place inside the mobile systems have the knowledge about their surrounding environment which includes information about the location, the parent and the child ambient and the processes executed. An ambient might use the information to adapt its behaviour according to the current situation.

Context-aware systems are a component of a ubiquitous computing or pervasive computing environment which offers any time, anywhere, anyone computing by decoupling users from devices. To provide adequate service for the users, applications and services should be aware of their contexts and automatically adapt to their changing environment. Context is defined by means of three important aspects:

- The location of the ambient i.e. where the ambient is currently situated
- The surrounding environment of the ambient i.e. who the ambient is with, and
- The available resources that can be used by the ambient to execute necessary processes [72].

Calculus of Context-aware Ambient (CCA) is a mathematical notation to represent ambient. It falls under the category of process calculus used to model the behaviour of context aware ambients [88]. CCA comes out with new constructs to enable ambients and processes to gain awareness of their environment. F.Siewe et al. in [88] have also proved that CCA succeeds in encoding π- calculus which is known to be a universal model of computation and provides a comprehensive linguistic support for modelling context-aware and mobile systems [89] [90].

The context includes information about the ambient, the location of the ambient, the surrounding ambients (i.e. parent, child and sibling ambients), the processes executed by the ambients, ambient preferences and ambient profiles to name a few. CCA has laid down a universal syntax to express and formalize any context-aware system [91]. The simplest entities that form the notation of CCA are names used to name the ambients where ambients may be users, locations or devices. The names of ambients in CCA are always written in lower-case letters. CCA is classified into four syntactic categories, viz. processes (P), capabilities (M), locations (α) and context expressions (k).

3.6.1 Processes

Notation	Description
P, Q	process
0	inactivity
$n[p]$	ambient
$(vn)P$	restriction
$P\|Q$	composition
$!P$	replication
$k?M.P$	context-guarded capability
$x(\tilde{y}).P$	process abstraction x

Table 3.1: Processes syntax of CCA

The process $n[P]$ designates an ambient named n where P is the process that describes its behaviour and characteristics. The pair of square brackets [and] outlines the boundary of that ambient, and everything that lies between those brackets is a part of the ambient behaviour. The process of an ambient connotes the functions and operations executed by the ambient. In CCA, the process $P|Q$ indicates that process P is operating in parallel with process Q. The process 0 does nothing and terminates immediately. The process $(vn)P$ indicates that the scope of the name n is restricted to the process P. Finally $!P$ is a process which can create a new replica of the process P. e.g. , $!P = P|!P$ [38].

A context expression specifies the condition that should hold true when matched with the environment of the executing process. The *context-guarded prefix* $k?M : P$ is a process that performs the capability M followed by the process P if, and only if the surrounding environment satisfies the condition which is defined by the context expression (k). The context-guarded prefix functions very much similar to the 'if-then' statement used on most of the programming languages. The 'if-then' statement means that if the condition specified by the 'if' statement is satisfied then the action specified by 'then' statement should be carried out. The use of context-guarded prefix is one of the two main mechanisms for context acquisition in CCA [38].

The second mechanism used for context acquisition is the 'call to a process abstraction' denoted by $x(\tilde{y}).P$. This symbol denotes the linking of the name x to the process P, where $(\tilde{y})$ is a list of formal parameters. A name x can be linked to more than one process in a different ambient. For example, a name x can be linked to a process P in ambient n and to process Q in ambient m [38].

A call to a process abstraction named x is done by the executing ambient by performing the capability $\alpha x < \tilde{z} >$ where α specifies the location where the process abstraction is defined and $(\tilde{z})$ is the list of actual formal parameters [38].

3.6.2 Location

As ambient has the capability to communicate with each other, the location of an ambient is an important parameter used as a reference by the communicating ambient. An ambient can exchange messages with a parent ambient, a child ambient or a sibling ambient. In CCA, the location α is denoted by '$\downarrow$for any parent ambient, by '$\downarrow$' for any child ambient and by '::' for any sibling ambient. More specifically, the parent of an ambient named $n$is denoted by '$n \downarrow$, the child of any ambient n is denoted by '$n \uparrow$' and the sibling of any ambient n is denoted by 'n ::'. CCA also allows an ambient to refer to itself (self-communication) using the symbol ϵ. Table 3.2 summarizes the location syntaxes mentioned in this section [38].

α	Description
'$\uparrow$'	any Parent
'$n\uparrow$'	Parent n
'$\downarrow$'	any child
'$n\downarrow$'	child n
'::'	any sibling
'n ::'	replication
ϵ	context-guarded capability

Table 3.2: Location syntax of CCA

3.6.3 Capabilities

Capabilities to send a message:

- $\langle a \rangle$: capability to send a message a to the parent
- $n \uparrow \langle a \rangle$: capability to send a message a to the parent ambient n
- $\uparrow \langle a \rangle$: capability to send a message a to any parent ambient
- $n \downarrow \langle a \rangle$: capability to send a message a to the child ambient n
- $\downarrow \langle a \rangle$: capability to send a message a to the sibling ambient n
- $n :: \langle a \rangle$: capability to send a message a to the sibling ambient n

Capability to receive a message:

- (a): capability to receive a message a from self
- $n \uparrow (a)$: capability to receive a message a from the parent ambient n
- $\uparrow (a)$: capability to receive a message a from any parent ambient
- $n \downarrow (a)$: capability to receive a message a from the child ambient n
- $\downarrow (a)$: capability to receive a message a from any child ambient
- $n :: (a)$: capability to receive a message a from the sibling ambient n

CCA specifies two basic types of capabilities for ambient: the mobility capabilities which enable an ambient to move around its environment, in and out and a few communication capabilities, also known as 'handshaking' capabilities [88]. According to the mobility capabilities, an ambient can perform the capability 'in' to move into a sibling ambient named n, and the capability 'out' to move out of its current parent ambient [38].

A process call $\alpha x \langle \tilde{y} \rangle$ indicates a process that is linked to x at location α. A process call takes place only if the corresponding process abstraction is available at the specified location. Using the capability $\alpha \langle \tilde{y} \rangle$, an ambient can send a list of names to a location α. To receive a list of names from a location α, an ambient needs to perform the capability $\alpha \langle \tilde{y} \rangle$ [38].

An ambient named n can be deleted by using the capability *deln*, but only under the condition that if the ambient is of the form $n[0]$, i.e. empty/ineffectual ambient. The capability *del* is considered as a garbage collector that deletes empty ambient which has no more computations to execute, i.e. $n[0]$ [38]. In terms of communication capabilities, an ambient can communicate with the parent ambient, the child ambient or with the sibling ambient. An ambient can send a message to another ambient or can receive a message from another ambient.

3.6.4 Context Model

An ambient is the basic structure that represents any entity in CCA. As mentioned before, an ambient has a name, a boundary and can be nested inside other ambient, which allows for a set of ambient to be represented hierarchically. A context model consists of nil, hole, location, parallel composition and restriction as the basic syntaxes that help to define the context expression (CE).

A hole, which is denoted by $\odot$, is a spatial context that represents the position of the executing process in the system. For example, let us say that a system is modelled by the process $P \mid n[Q \mid m[R \mid S]]$. So, the context of the process R in that system is $P \mid n[Q \mid m[\odot \mid S]]$, and that of the ambient named m is $P \mid n[Q \mid \odot]$, and that of ambient n is $P \mid [\odot]|m[R|S]$ [38].

In Table 3.3, E represents the context, n stands for names and P denotes the process. The context $\mathbf{0}$ represents an empty (ineffectual) context, which does not provide any information. The hole $\odot$ indicates the position of a process in that process's context. The context $n[E]$ signifies that the context E is the internal context of the ambient n. The context $P|E$ indicates that the process P operates in parallel with the context E. The last context $(n)E$ refers to a name restriction, in which the name n is restricted to the context E [87].

3.6.4.1 Context expressions

A context is any information that can be used to characterise the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves. A context expression (CE) is a predicate that states the condition the environment of the executing process must meet. Context Expressions will be dealt with in detail in chapter 5, '*Reserving the parking space formalisation*'.

The CE True always holds, whenever it indicates the value true. The CE $\bullet$ holds for the hole context only, i.e. it represents the process evaluating that context expression. For the CE $n = m$, it applies only if the names n and m match exactly. Such CE is important for checking if two messages have got the same content or

K ::=	Context Expression
$True$	true
$n = m$	name match
$\bullet$	hole
$\neg k$	negative
$k_1 \| k_2$	parallel composition
$k_1 \wedge k_2$	conjunction
$n[k]$	location
$new(n, k)$	revelation
$\oplus k$	Spatial next modality
$\diamond E$	somewhere modality
$Ex.k$	existential quantification

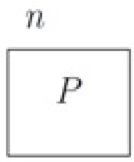
Table 3.3: Context expressions syntax

not. The first order logic operators $\neg$ (negation), $\wedge$ (and) and E (there exist) hold valid in CCA and expand their standard meaning to CEs [38].

The CE $k_1|k_2$ holds for a context if that context is a parallel composition of two contexts such that k_1 holds for one and k_2 holds for the other. The CE $n[k]$ holds for a context if that context is an ambient named n such that k holds inside that ambient. The CE $new(n, k)$ holds for a context if that context imposes a restriction of the name n to another context for which k holds true. The CE $\oplus k$ holds for a context if that context has a child context for which k holds true. The CE $\diamond k$ holds for a context if there exists somewhere in that context a sub-context for which k holds. The operator $\oplus$ is called as spatial next modality and the operator $\diamond$ is known as somewhere modality [87] [88].

3.7 Graphical Representation of an ambient

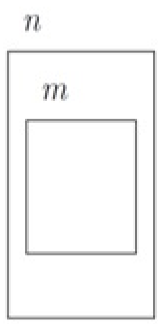
Recalling the definition of ambient from section 3.1, an ambient can be defined as a bounded region where some computation takes place. An ambient is an entity used to describe any object or component (e.g. person, process, device, location, etc.) in a system [86]. As an ambient is identified through a name, it is present in some environments and executes some processes at a given time, any ambient can be represented graphically as follows:



Where

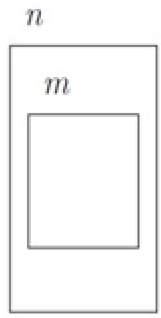
- n: is the name of the ambient.
- P: is the name of a process that represents the behaviour of the ambient n.

The graph mentioned above is textually read as n[P], where n indicates the ambient's name and P is the process (computation) describing the behaviour of the ambient at that point of time. For example, a Smart phone can be modelled as Sphone[P] , where Sphone is the ambient name of the Smart phone and P encompasses the functionalities of the Smart phone. As mentioned in the introduction, an ambient can also have other ambients inside its boundaries, allowing a set of ambients to be represented hierarchically. For example, the following figure shows an ambient 'n' having an ambient 'm' inside [38] [88].



When two ambients act in the same environment and share a context between them, then the two ambients are said to be in relation with other ambients and there are three possible relations among ambients: parent, child and siblings. These relations are described graphically as follows:

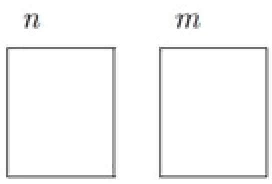
- An ambient n is said to be a parent to an ambient m if the ambient m is contained inside the ambient n, as depicted in the following graph.



  Here, the ambient m is said to be a child to ambient n.

- An ambient n is a sibling to an ambient m, if both m and n share the same parent.

  Ambient n is a sibling to ambient m as represented in the following graph:

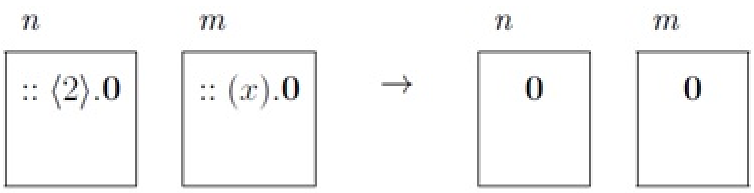


  The symbol '::' represents a sibling ambient. The graphical representation of ambient is a fundamental step in defining the behaviour (formalisation) of any context aware system. The formalisation is done using CCA, so it is important to mention some of the CCA capabilities that we are going to use while working with ambients in this section.

- The first CCA capability used is the 'handshaking' process.

An ambient is capable of communicating (send/receive messages) with other ambients in its surroundings. Those ambients can be the parent, child or the sibling ambients. This course of action of exchanging messages is done by using handshaking. For example, if an ambient n sends a message, say 2, to an ambient m then the ambient m also has to be able to receive such a message from ambient n to complete the process of message exchange. This mechanism of completion of message exchange between the ambients is called 'handshaking'.

Symbolically $n :: \langle x \rangle$ represents the capability to send a message 'x' to the sibling ambient 'n' and $m :: (x)$ represents the capability to receive a message 'x' from the sibling ambient 'm'. In the following figure, ambient 'n' sends a message '2' and the sibling ambient 'm' receives the message, as characterised by the symbols [38] [88].

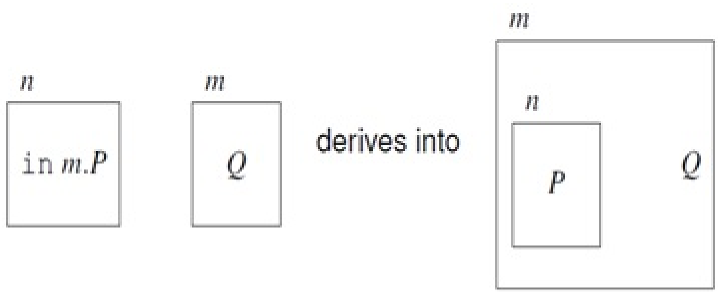


An ambient can also perform the procedure of handshaking with a parent ambient or a child ambient, using the symbols $\uparrow$and$\downarrow$, respectively. Symbolically, $n \uparrow \langle x \rangle$ represents the capability of an ambient to send a massage 'x' to the parent ambient 'n'. Similarly $m \downarrow (x)$ represents the capability of an ambient to receive a message 'x' from the child ambient 'm'.

- The next CCA capability used is the 'mobility'.

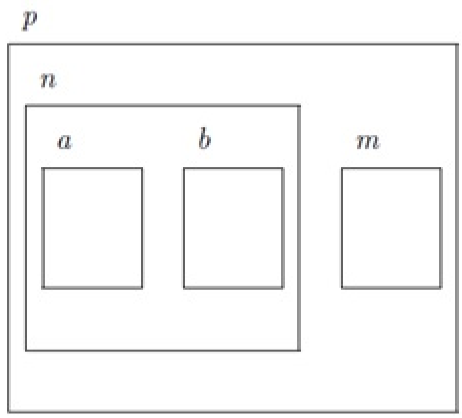
  An ambient has the ability to move in the surrounding region of its environment. An ambient is allowed either to move into a sibling ambient, or out of its parent ambient. As a default process, if an ambient moves around its environment, then the entire child ambients of this ambient necessarily have to move with it.

In CCA, an ambient has two mobility capabilities: in and out. An ambient can perform the capability 'in' m to move into a sibling ambient m as shown in the following graph. Here the ambient n moves into the sibling ambient n.



Similarly, an ambient can perform the capability 'out' to move out of its current parent as shown in the graph below. Here the ambient n moves out of the sibling ambient m.

In terms of context-awareness, an ambient has the ability to sense its surroundings (other ambients) in the environment. Basically, the knowledge of the parent ambient, the child ambients and the sibling ambients around a specific ambient identify the environment of that ambient. The ambient is supposed to be located inside the parent ambient. For example, in the graph below we say that p is the location of the ambient n since ambient n is located inside the parent ambient p. Similarly, the environment of the ambient m in the same graph includes the ambients p, n, a and b [38] [88].



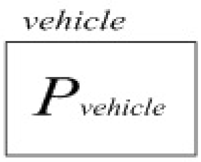
Now that we have a clear understanding of what an ambient means and the characteristics that give it the potential to model any entity in a system, therefore in the next section, we present the basics of the context-aware on-street parking system in CCA. We will begin with identifying the ambients, their environment, and the communication and mobility between various ambients.

3.8 Modelling the System in CCA

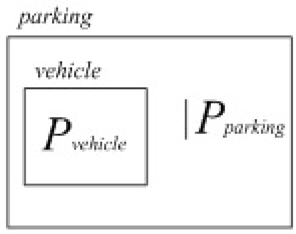
As mentioned in the previous section, CCA is a process calculus based on the notion of ambients. An ambient is aware of its context/environment. The environment of an ambient is defined by other ambients around it. Ambients are the basic structure used to model the entities of a context-aware system. As we know, an ambient has a name, a boundary, a set of processes to execute, and can be a part of other ambients or can contain other ambients. An ambient is also qualified to move from one location to another by performing the mobility capabilities of 'in' and 'out.' Therefore the structure of CCA expressions at any time consists of a hierarchy of nested ambients. As the process begins to execute, the hierarchical structure can show up some changes [88].

As stated above, CCA is used to model ambients in terms of the processes executed by the ambients, their location, mobility and communication capabilities. For modelling in CCA, an ambient first has to be modelled graphically, displaying all the neighbouring ambients in the context. So in this section, we demonstrate the graphical model of the context aware on-street parking system.

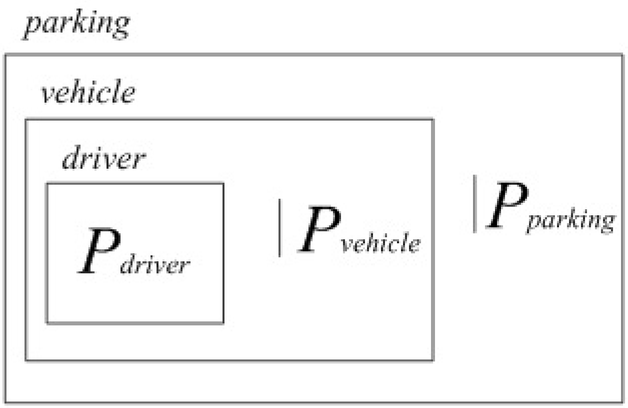
The ambients in our proposed system will be the parking zones, vehicles and drivers. The graphical modelling an ambient with respect to its context, its mobility and communication capabilities is presented as follows. We begin with modelling the vehicles as an ambient named vehicle. The context of the vehicle is represented by the vehicle's location, the neighbouring vehicles, surrounding environment of the vehicle, available parking resources etc.



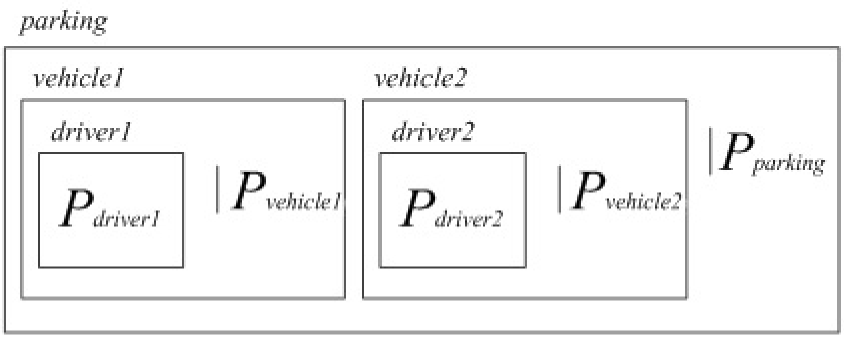
Where $P_{vehicle}$ is a process specifying the functionality of the ambient vehicle. CCA models an ambient considering the location of the ambient where location of an ambient is modelled as a parent ambient and the ambient that is located inside the parent ambient is modelled as a child ambient. For example, if the vehicle is parked in a parking zone, then the location of the vehicle can be modelled as follows, where the ambient parking (parking zone with a fixed IS) represents the location of the ambient vehicle:



The next graph represents an ambient driver located inside the ambient vehicle. The ambient vehicle is itself located in another ambient parking.



The parking zone may contain more than one vehicle denoted by vehicle1, vehicle2 and so on. The environment of each vehicle is composed of every ambient around that vehicle (parking, vehicle1, driver1, vehicle2 and driver2 etc).

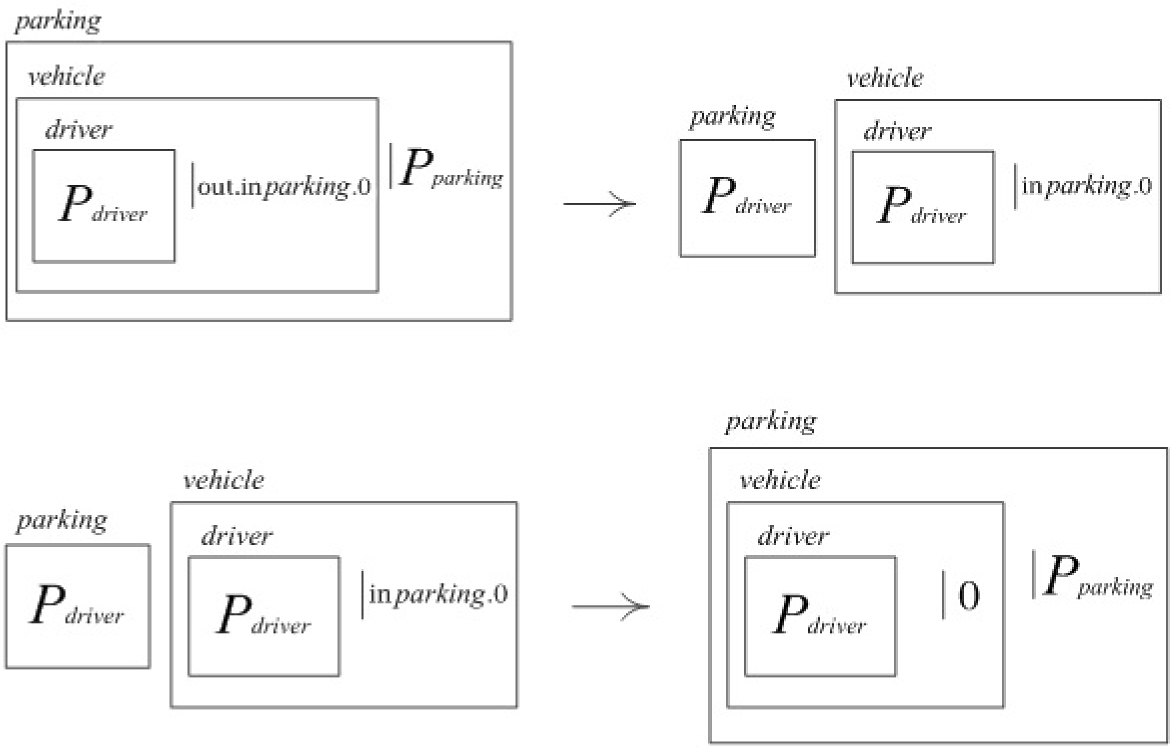


Syntactically, this graph is represented as follows:

$$\begin{aligned} Parking \quad & [\ Vehicle1\ [\ Driver1\ [\ P_{Driver1}\ ]\ |\ P_{Vehicle1}\ ] \\ & |\ Vehicle2\ [\ Driver2\ [\ P_{Driver2}\ ]\ |\ P_{Vehicle2}\ ] \\ & |\ P_{Parking}\ ] \end{aligned}$$

An ambient vehicle can be mobile; it can move around within its environment. In CCA, as already mentioned, 'in' and 'out' are two essential mobility capabilities that enable ambients to move in or out of a particular location. The capability 'in' is used by the ambient vehicle to move inside a sibling ambient. An ambient can use the capability 'out' to move out of its parent ambient.

As ambients can be represented hierarchically, when an ambient moves around in its environment, all its child ambients move with it. For example, the vehicle1 can move out of parking and then move in again by performing the capabilities 'out' and 'in' respectively, noting that the driver ambient is a child ambient to vehicle1 as the driver is located inside the vehicle.



3.9 Communication Capabilities

Let us recall the 'handshaking' process where an ambient uses its communication primitives to send and receive messages from neighbouring ambients. In the context-aware on-street parking system, if the driver sends a parking request from the vehicle to the parking zone, then it is a parent-child communication. If the driver sends a message to a neighbouring vehicle that is within the range of the desired parking IS, then it is a sibling-sibling communication (i.e. both the vehicles have the same parking IS as the parent ambient).

3.10 Summary

The chapter introduced the concept of the computational model which is based on the notion of an ambient. The characteristics of ambient were defined and it was explained why an ambient is the perfect candidate to represent any entity or object in a context-aware system. The two important capabilities of ambient were highlighted: the communication primitive and the mobility primitive which allow the ambient to send/receive information to/from other ambient and to move in/out of other ambient. The chapter also presented an introduction to two modelling methods of ambient: mathematical and graphical.

The next major part of the chapter was the introduction to Calculus of Context-aware Ambient (CCA). CCA was defined as one of the process calculus methods that provide a mathematical notation suitable for formally specifying an ambient. The sections that followed illustrated the syntax of CCA in terms of processes and capabilities, explained the informal semantics and illustrated some examples. These syntax and semantics of CCA help to describe the context model and context expressions of communication and mobility taking place between ambient.

In the last part of the chapter, we presented a few graphical models of context-aware on-street parking system in CCA. In the next chapter, we will present a novel three-tier VANET based architecture of our proposed IS based context-aware on-street parking system.

Chapter 4

On-Street Parking System Architecture

Objectives:

- To propose a novel context-aware intelligent parking system architecture in VANET
- To define the components of the proposed system architecture
- To explain the three tiers used in the context-aware intelligent parking system
- To describe the technique for requesting on-street parking space

4.1 Introduction

VANET is a new technology that treats moving vehicles as nodes in a network. Using the DSRC, vehicles in VANET can communicate with each others or with RSU which in turn can communicate with the entire network using any of the available communication medium. As shown earlier in chapter two, these types of communication can be classified into three domains: in-vehicle (i.e. communication between the AU and the OBU), ad hoc (i.e. V2V or V2I) and the infrastructural domain (i.e. RSU communicate with any host such as internet) [29] [36].

Context-aware systems are those system which have the ability to sense the surrounding environmental context using different kinds of sensors, reason about this context and then provide appropriate services to users according to current context. Three subsystems have to be involved in any system to be context-aware which are sensing, reasoning and acting subsystems [58].

This chapter present a novel context-aware intelligent parking system architecture in VANET, with a thorough explanation for its components. The mechanism for finding a free on-street parking space, based on the proposed architecture is also illustrated. The architecture utilises the ad hoc and the infrastructural domains of VANET and is based on the concept of a context-aware system, in which it comprises three tiers which are vehicle with OBU, IS and ISC, and constitutes three subsystems: sensing; reasoning and acting subsystems which denote the three subsystems of the context-aware system. Using this proposed technique the architecture has been built to locate and reserve a parking area in the desired destination based on the request of the driver.

4.2 Finding On-street Parking Technique

This section, using an automated system presents a novel technique for requesting a parking space in a specific area based on driver's preferences, estimated time for arrival, vehicle preferences and several other inputs. The flowchart in Figure 4.1 explains the process of requesting a parking space on a specific area. The driver sends a request message which contains information about the driver (i.e. disabled or non-disabled), duration of stay in the parking place, parking type (i.e. free of charge), the area in which the parking place is needed, vehicle size and the current location of the vehicle. These information can be obtained from sensors, including physical and logical sensors, such as ultrasonic, driver profile, vehicle profile, driver preferences and the GPS.

The request will be sent via the DSRC wireless device equipped within the vehicle to the nearest IS using the wireless access technology provided by VANET. The RSU forwards the request to the ISC in order to find a parking space according to driver's request. In the ISC, the location calculator will match the request received from driver with its database depending on certain criteria, the database contains information provided by all ISs in the area (i.e. available parking type, permit duration, available space and the address of parking zones).

If there is any match for the request, the IS will send back a message to the IS which in turn will forward it to the vehicle, the message contains information about the place of the parking zone and the place in which the vehicle has to park. In the case where there is no match for the driver request, the alternative space calculator will find the alternative options with respect to the requested information, and a message with a set of options will be send back to the IS in order to sent it back to

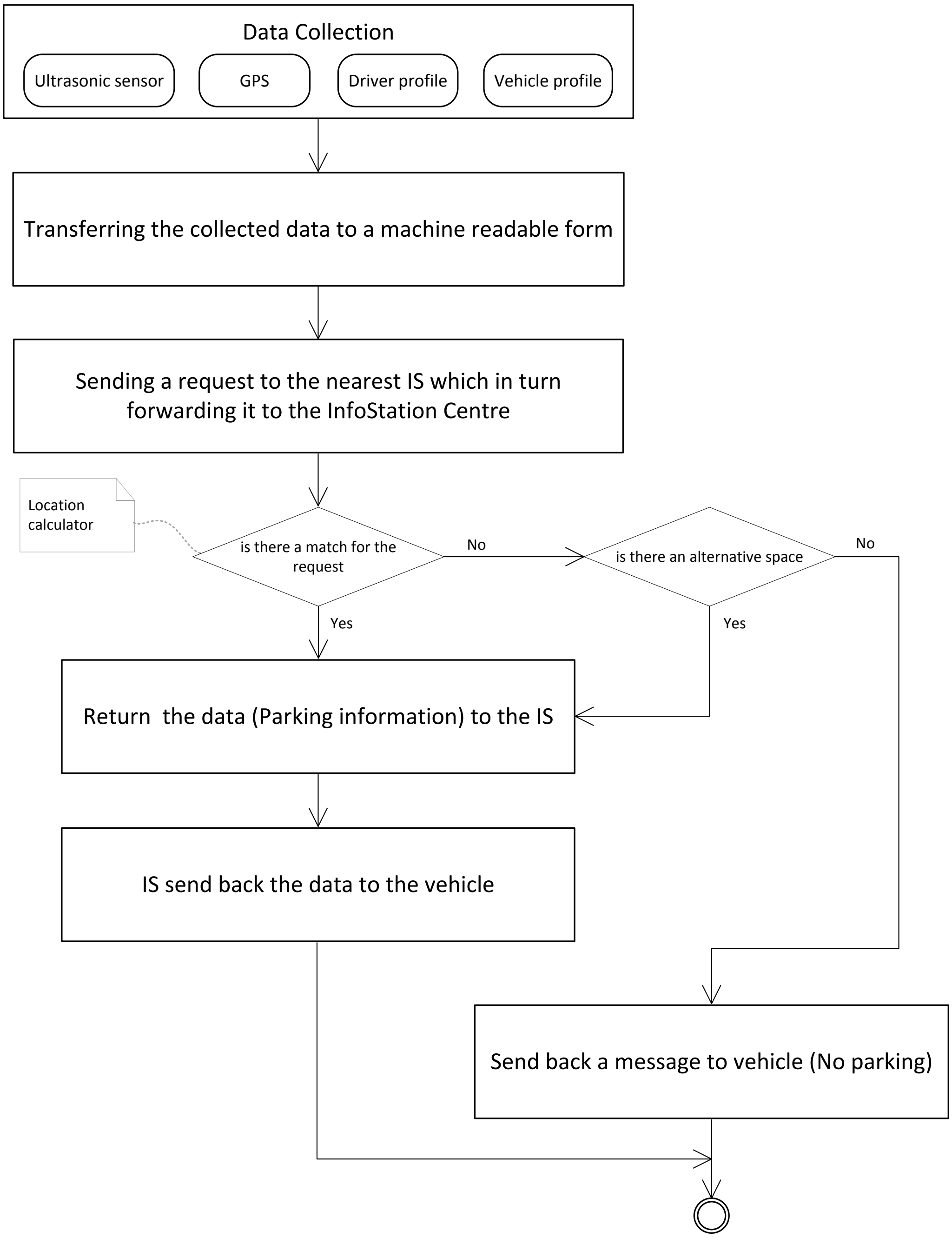


Figure 4.1: Finding on-street parking technique

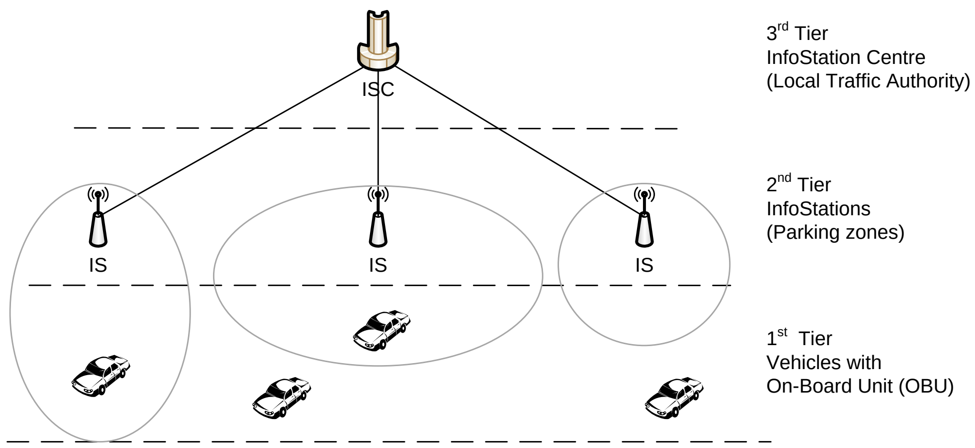


Figure 4.2: Intelligent parking system hierarchy

the vehicle.

Based on the above technique, the system has been divided into three separate tiers consisting of the building blocks shown in Figure 4.2. In the following section, each tier will be illustrated in detail:

- *FirstTier*: This tier consists of vehicles equipped with OBUs which allows each vehicle to communicate with the parking system ISs via DSRC provided by VANET in order to send request message. These vehicles can be distributed at any given point in time such that some of the vehicles fall in the range of an IS, while others might not fall in the range of any IS. vehicles that fall in the range of the IS are adequate enough to send their requests to the IS directly, while those out of the range of the ISs, have to make their request through other neighbouring vehicles.

  The vehicle's OBU is equipped with a Graphical User Interface (GUI) where the driver can interact with the system in terms of sending request and receiving information for the desired parking. Messages sent by the driver and those received back are all displayed on the OBU's screen. The OBU is defined as a wireless access in vehicular environments (WAVE) device that can operate when in motion and supports information exchange with IS or other OBUs [30].

The vehicles also have two types of sensors mounted on them. The first type of sensors are mounted on the bumper of the car (on the sides of the car), which are called the 'surround sensors'. These sensors are used to calculate the distance of the vehicle with the nearby vehicles in the parking zone in order to check if the distance between the vehicles is maintained as per the requirements. The second type of sensors is called the GPS sensors that are mounted on the top of the vehicles, which sense the location of the vehicle. The information sensed by both the types of sensors is displayed on the OBU.The OBU may also include additional resources such as network interfaces and security devices [30].

- *SecondTier*: This tier constitutes ISs, which act as RSUs for the parking zone. The IS is a node that is stationary while in operation and usually mounted along the roadside, and it hosts an application that provides required service to the OBU installed in the vehicles [30]. An IS contains a detailed information about the parking zone and its policies (including information about the available space, type of parking zone, location and the permit time of stay). The suggested parking place may be free or might be suitable for short stay only etc. The IS provides an effective way to facilitate the parking services for mobile vehicles on the road. The vehicles are already equipped with an OBU. When the vehicles are looking for parking space, the OBU sends a request to the IS.

  As described in the subsequent sections, the message sent by the OBU contains detailed information about the required parking place, it also contains all the information about the parking preferences that the driver is looking for. The IS receives the message from an OBU and forwards it to an ISC in order to match the request with the database stored in the ISc, and then relays an appropriate reply back to the vehicle. This reply is again shown to the driver on the display of the OBU. This communication between the vehicle OBU and the IS is provided through the DSRC, regardless of whether the communication is V2V or V2I. The ISs assumed to support the standard VANET connection DSRC in order to provide sufficient access for vehicles.

ISs unit along the parking zones contains the latest parking details and polices an it usually synchronise this information with the ISc in order to keep it up to date [39] [87]

- *ThirdTier*: The ISC (the core of the IS system) is considered as third tier in the proposed system. It controls the ISs and is responsible for matching the driver's request and finding either the appropriate parking place or alternative options with regards to the received request. The ISC is synchronised and is updated with the ISs, this includes parking zones (i.e. vacancy of the parking, parking policies, payments process, etc.). It is preferred that the ISs across the system are connected to the ISC with a wireless communication, but in remote and physically difficult terrains where installing internet towers and antennas might not be feasible, it is proposed that direct cable is used to provide an efficient, reliable and secure communication.

Dividing the system into three tiers makes the process of searching and reserving a parking space methodical, and facilitates the transformation of requests from vehicles to ISC. In the following sections, the proposed architecture which is divided into three blocks based on the above tiers will be illustrated in detail.

4.3 The Architecture

This section enlists the principal parts of the architecture and their functions. Figure 4.3 depicts the proposed architecture of the intelligent parking system with the main components listed as blocks, and their interaction represented by horizontal and vertical connections. Once the basic components of the parking system have been identified and listed, we will then explain the full details of each of the components.

In fact, each component of the system is accomplished to provide more than one service to complete the process of the intelligent context-aware parking system. For instance, the vehicle in this system can act as a client requesting for a free parking space and at the same time it can also act as a mediator node forwarding other vehicles' request, which are not able to communicate with the IS directly.

Full definitions of the parking system information, including parking policies and request message will be explained in Chapter 5.

The architecture aims more at explaining the overall concepts involved in the intelligent parking system, than illustrating the details of the implementation. So, first the contents of the system components will be explained in details starting from the vehicle contents, the IS contents and the ISC contents. Later, the steps carried out by each and every block in the execution of the parking reservation process, the steps involved in enabling the communication between the nodes will be explained along with the proposed system technique.

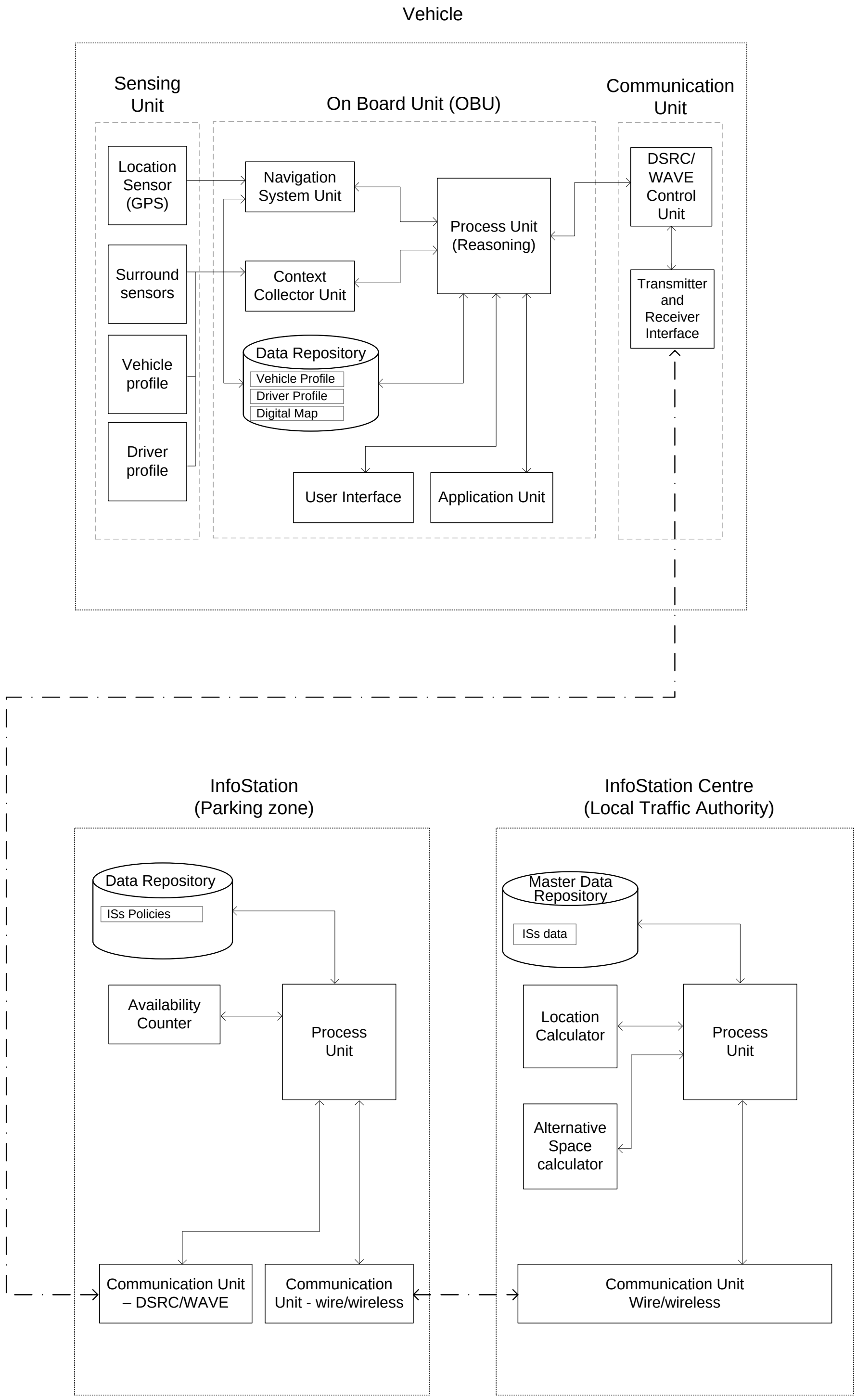


Figure 4.3: Intelligent parking system architecture

4.3.1 Vehicle block

The initiation of the parking space request always starts from the vehicles side (i.e. the service user). This request usually involves different devices on the vehicle, together with the driver interaction to process and forward the parking request. In fact, to demonstrate all the tasks carried out by the vehicles in requesting and reserving a parking space we need to identify the key process and outline all the operations involved in the parking reserving mechanism.

4.3.1.1 The Sensing units

In order for the system to be based on accurate and actual data resources we must take into account the reliability of its functionality. Using different types of sensors (including physical and logical sensors) will provide the user with usable data to assist the requesting process. Considering the way the data is captured using sensors, the context can be captured in three different ways: physical, virtual and logical sensing [92]. Based on the above said methods of sensing, our sensing system will include the following two major types of sensors:

- Surround sensors: The surround sensing system (vehicle parking aid sensors) is a physical sensor mounted on the sides of the vehicle to detect the parking environment. When a vehicle arrives at a parking zone, the parking policies state that the vehicle should maintain a gap of 120cm with another vehicle, on any one side of the vehicle. The surround sensors sense the gap and help the vehicle to adjust the gap to 120cm. These sensors utilise an ultrasonic wave emission technology to detect the other vehicles within a pre-specified distance. The sensors mounted on the vehicles emit ultrasonic pulses, and these pulses get reflected if they hit an obstacle. Depending on the time taken by the pulses to reach back to the vehicle, the distance is calculated [52] [93].

With respect to our system, the free parking spaces (in a parking zone) will be identified based on gathered information from the parked vehicle sensors. The surround sensors will capture the front and rear distance information and pass it to the OBU to process the data and figure out the vehicle position and free spaces as well (on the parking zone). Once the vehicles complete the parking process; the information will be passed to the parking zone IS to keep it up to date with information regarding the free and occupied parking space.

- Location sensor Modern vehicles nowadays came equipped with a location system detector (sensor), commonly known as a Global Positioning System (GPS) mounted on the top of the vehicle. It is a satellite-based system which provides a continuous positioning and timing information, anywhere in the world under any weather conditions [94].

  This positioning information is very useful for the vehicles as it helps the IS to allocate the right parking zone to the vehicle in the right time. The process of finding a vehicle position starts from the GPS receiver. It receives the satellites signals and then passes it to the navigation system to figure out the actual vehicle position based on the standard geographic map coordinates.

- Vehicle profile: This profile contains information pertaining to the vehicles description such as vehicles make, type and size. In fact, the ISs will use the vehicle registration number to find out all information about the vehicle. This important information is needed by the ISs to allocate the right parking space for the right vehicle.

- Driver profile: This profile contains all needed driver's information. The driver's profile proposed to contains the official information that are registered on the driver license such as license identification number, driver name, medical conditions and any additional information that may used by the parking system. In terms of security, this information proposed is to be registered with an official organisations such as the Driver and Vehicle Licensing Agency (DVLA) in the UK.

4.3.1.2 The on-board unit

An On-Board Unit is a device in the vehicle having communication capabilities (wireless and/or wired) allowing the Application Unit (AU) to executing a single application or a set of applications depending on the user priorities [28]. The message sent by the driver and those received back are all displayed on the OBU's screen, which is fitted with a GUI is will shown later.

The OBU allows the vehicle to utilise both, the in-vehicle domain and the ad-hoc domain. The ad hoc domain will allow the vehicle to communicate with ISs (parking zones) directly (or via other vehicles) along the road using the DSRC wireless communication provided by its OBU. The ISs may allow OBUs to access the infrastructure and, consequently, to be linked to the network resources [28] including the parking services. Thus, the components of the OBU and the major functions performed by them can be summarized as follows:

- User Interfaces: The OBU employ a mixture of UIs to allow the driver to interact with the vehicle in different ways. It may provide visual displays, buzzer alert, speech enunciators and keypad interfaces to reflect application actions. The OBU may also include additional resources such as network interfaces and security devices. Our proposed architecture is designed in such a manner so as to enable a wide range of applications to be supported by an OBU at the lowest possible cost [30].
- Data repository: The data repository is an essential component on OBU. It

will store all data that may need by the system such as vehicle information, driver information and a digital map. This component also stores the driver and the vehicle profiles.

- Digital map: The OBU data repository will contain the preloaded digital map of the roads and the transport informant to assist the navigation system. This map and the GPS will provide the direction, arrival time and the full journey route to the proposed system. The map related functions of a navigation system are: map matching (to correct the positioning), route calculation (to determine the best route), route guidance (to give turn-by-turn instructions to the required destination. The digital map has to fulfil quality criteria such as correctness, completeness, up-to-datedness and accuracy.

  For route guidance these criteria are highly severe and the accuracy strictly needs to be adhered to. The data must also contain a high volume of traffic attributes (one-way streets, banded turns, complex intersections, etc.) which are necessary to give the driver the correct instructions next turn. This is the essence of the navigable digital street map [95] [96].

- Application Unit (AU): The Application Unit is a device on OBU to execute a single or a set of applications using of the OBU's communication capabilities. This is the primary component which enables the OBU and the IS to have meaningful communication with each other. The Author in [28] has proposed the AU to be an integrated part or portable device connected (wireless or wired) to OBU, while in this architecture the AU is proposed as an integrated feature inside the OBU to account for the ease of system use and operability and to ensure the compatibility with the hardware as well.

- Context collector unit: This unit is responsible for transforming the data collected by sensors into a machine processable form (data received from surrounding sensors, driver profile and vehicle profile). It will maintain the information sensed from the surroundings in an updated and ready to use form in order to be retrieved by an application.

- Navigation system unit: The navigation system unit (NV) is an internally installed device on the OBU which has the independency and rich navigation information including the position, velocity and altitude relate data [97]. It utilises the preloaded digital map (on the OBU data repository) and the location sensor (GPS) to obtain the required information from the parking system.

- Processing unit (Reasoning): The processing unit is the core of the OBU, and it controls and coordinates the operation of the other OBU components. The information from all the components mentioned above is fed to the processing unit, and the processing unit is responsible to perform reasoning about certain contextual information, that has been acquired directly from the sensors such as the surround sensing.

  As a matter of fact, the processing unit on the OBU is responsible for implementing the parking request services in terms of process the data and send it to the IS. The starting point of the parking request services is from the vehicle driver request (from the UI), upon receiving which the processing unit of the OBU assembles all the needed data (contextual information, digital map) and then sends this request to the specified IS (parking zone). The processing unit is connected to the communication unit where it sends all the required information to the IS using the dedicated short range communication.

- Communication unit: The communication unit consists of the DSRC control unit and the transmitter and receiver interface. The communication unit uses the DSRC spectrum to send the parking request data to the IS, and receive acknowledgements back from the IS. The DSRC is used by the communication unit because DSRC is a one-way or a two-way short-to-medium range wireless communication channel, specifically designed for automotive use and is commonly divided into sub layers [28].

4.3.2 IS block

IS is a comparatively new concept developed ideally to support "many time, many where" wireless data services. It allows mobile terminals (i.e. vehicles) to communicate to ISs with variable data transmission rate and obtain the optimised throughput [98]. The IS is defined as a comprehensive information acquisition, processing and delivery system. It is a node that is stationary while in operation and usually permanently mounted along the roadside, it hosts an application that provides service to the OBU mounted on the vehicles [30].

In the IS block, the major components are the policy matching and the availability counter components. The proposed architecture has been designed based on a network infrastructure using the ISs providing parking services, through wireless access points typically deployed at the key points within a parking zone [99], for the nearby vehicle drivers equipped with OBU.

The proposed IS in our architecture contains the following components:

- Data repository unit: The data repository of each IS keeps an archive of parking zone policies and the available parking space. These polices may vary from one zone to another, depending on the parking deployment. It is managed remotely (via wire/wireless connection) by the ISC to be created, updated, and deployed. The ISs data repository will maintain the up to date policies and the available parking space on the parking zone. This updated data needs to be ready all the time with the IS to meet the parking request from the number of vehicles.

- Availability counter unit: After gathering the information from the vehicle surround sensors on the parking zone, this unit will monitor the parking zone and figure out the free spaces based on the sensors information. As the number of vehicles requesting for a parking is huge at any given moment of time, and as the IS has to process a large number of transactions simultaneously, therefore this information has to be up to date and synchronised with the ISc.
- Process unit: This unit represents the processor of the IS controlling all the tasks in the IS and coordinating all the activities between the units inside the IS. It is responsible for forwarding the request received by a vehicle to the ISs and then send back the message to the vehicle depending on the matching or mismatch of the ISs's policy, and to update the ISC database.
- Communication units: These units are responsible for the communication with the vehicle or with ISs using the available wireless access technology.

4.3.3 ISC block

The third block in this architecture is the ISC. The main functions of the block is to control all IS in the system and to find the requested or the alternative parking places. It is responsible for updating and synchronising the parking service information to keep all the ISs up to date. As proposed, this part will be assigned to the Local Traffic Authority who is responsible for deploying and maintaining the parking zones and its policies. The main components of this block are as follows:

- Master data repository unit: The master data repository (MDR) contains up to date information about all ISs across the parking system (i.e. serial number, location, policies and parking lot size). This data will help the system to speed up the query process at the ISC side. This information also helps to maintain synchronization between the ISC, IS and the vehicles.

- Location calculator unit: This unit is responsible for finding the available parking space with regards to the request received by the vehicle, this process is carried out by matching the request with the database which contains information about all parking zones. The location calculator helps the ISC to locate an available parking space. In the case where there is no match for the request, the processor will send a signal to the alternative space calculator in order to find a list of place and send it back to the driver.

- Alternative space calculator: This unit is responsible for finding alternative spaces for driver with respect to his/her original request in the case where the location calculator fail to find the requested space.

- Process unit: This unit represents the processor of the ISC, which is responsible for controlling all tasks in the ISC and coordinating all activities between the units inside the ISC. It is also responsible for sending back the message to the vehicle via the IS.

- Communication unit: This unit is responsible for the communication between the ISC and other ISs in the system, using either wire or wireless communication technology.

4.4 Summary

In this chapter, a novel intelligent parking system architecture for VANET, based on the concept of a context aware systems has been introduced. A new technique for reserving on street parking based on the proposed architecture has been presented in order to make the parking reservation system intelligent and comfortable. The architecture has been divided into three blocks: vehicle, IS and ISC blocks, each of which have dedicated functions. Classifying the architecture into different blocks leads to effective and well defined communication between the vehicle and the IS.

According to the proposed architecture, the OBU installed in the vehicle holds all the necessary information that needs to be sent to the IS in order to request a parking space. The IS forwards the request to the ISC, which in turn knows the parking policies, it matches the information provided by the OBU with the IS policy to decide whether the requested parking is suitable for the vehicle as per the vehicle profile and the driver profile, and then will return information about the desired parking or the alternative one. The following chapter (Chapter 5) will illustrate the formalisation of the proposed architecture in detail.

Chapter 5

Reserving a Vacant Parking Space Behaviour

Objectives:

- To introduce the parking space reservation activity diagram
- To explain the mechanism of reserving a parking space
- To formalise the parking space reservation system in CCA

5.1 Introduction and Motivation

The architecture of the IS-based context-aware on-street parking system mentioned in Chapter 4 represents the mechanism of reserving a vacant parking space. In this chapter, a novel approach for V2I communication for locating and reserving a suitable parking space is presented. The approach mentioned in the chapter uses GPS and the road map system through which vehicles acquired knowledge of their current position and the direction of motion and therefore request for a parking space at the desired destination [100]. With the proposed approach, vehicles can book a parking space while travelling to the destination well before arriving. The proposed approach aims at reducing the 'search time' that is currently invested in locating a suitable parking space, freeing up the drivers' frustration and also cutting down the emissions that take place when a vehicle searches for a place to park [25].

In the proposed approach, the parking request message (RM) is sent to the IS using DSRC wireless communication protocol [34] [101]. The DSRC protocol performs node to node communication based on the IEEE 1609 standards of Wireless Access in Vehicular Environments (WAVE) family. This introduces a uniform communication platform between the IS and a number of vehicles. It comprises of four standards, one of which administers the services related to comfort applications. To enable V2I and V2V communication each vehicle needs to be equipped with an OBU [100].

The process for modelling the context aware system is known as the Calculus of Context Aware (CCA). Congruency, mobility and context-awareness are the main characteristics of the calculus and it is for this reason it has been selected to formalise and verify the proposed system. The virtual place where the calculations take place is referred to as the 'ambient'. Furthermore, these virtual places can be mobile as well as containing other ambients known as 'child ambients'.

5.2 The Behaviour of the On-Street Parking System

This section presents a time-efficient and cost-effective technique for reserving a parking space in a specific area based on the driver's preferences, driver's profile, vehicle's profile and several other pieces of information. The activity diagram in Figure 5.1 explains the process of requesting a parking space which is divided into three tiers, starting with the vehicle. The first tier represents the processes related to the vehicle combined with the OBU. The second tier represents the processes related to the IS covering a parking zone. The third tier represents the processes related to the ISC which is the local traffic authority. The three tiers co-operate in unison while executing the processes related to the parking reservation system.

The driver sends a RM which contains information about the driver (e.g. disabled or non-disabled), duration of stay in the parking place, parking cost (e.g. free of charge or payable), the area in which the parking place is needed, vehicle size and the current location of the vehicle. This information can be obtained from sensors, including physical and logical sensors, such as ultrasonic, driver profile, vehicle profile, driver preferences and the GPS. The request will be sent via the DSRC wireless device equipped within the vehicle to the nearest IS using the wireless access technology provided by VANET. The IS forwards the request to the ISC in order to find a parking space according to driver's request.

The Master Data Repository (MDR), located within the ISC contains policies

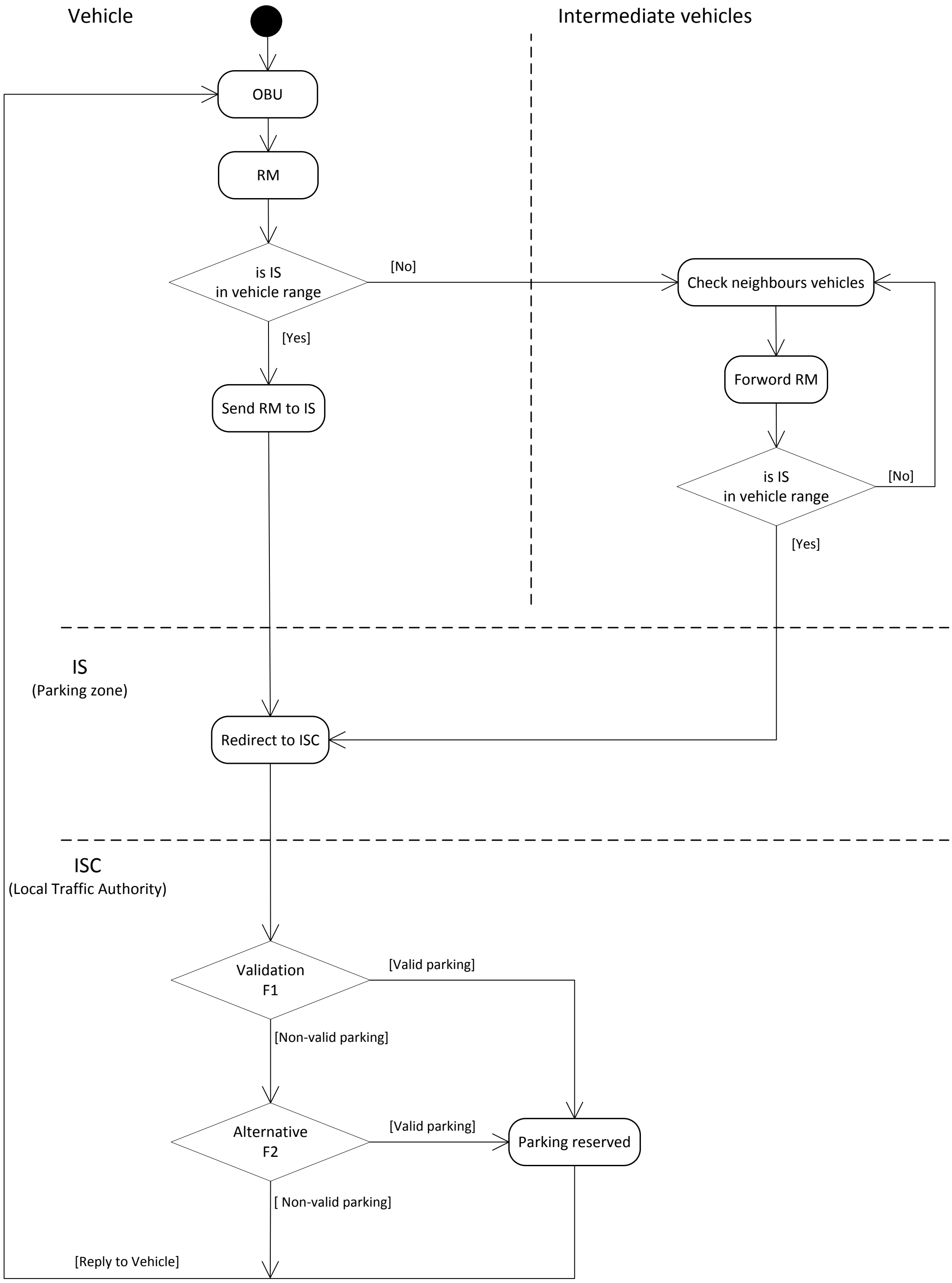


Figure 5.1: Finding on-street parking activity diagram

and information provided by all ISs in the area (i.e. available space, parking scheme, permit duration and the address of parking zones). Here, the location calculator will match the request received from the driver with its database depending on certain criteria. If there is a match for the request, the ISC will send a message back to the IS which in turn will be forwarded to the vehicle, containing information about the parking space available.

In the case where there is no match for the driver's request, the alternative space calculator will find alternative options with regards to the requested information. An appropriate alternative will be sent back to the IS which in turn will be forwarded on to the vehicle. The rationale behind the system by selecting an alternative parking zone for the driver without interaction, allowing them to focus their attention on driving as opposed to decision-making.

5.2.1 The Structure of the Parking RM

The parking request message contains a number of parameters to help the parking system to allocate the appropriate parking space for the right vehicle and driver. The following Table 5.1 describes the message contents in detail:

Parameters	Description
DE	Destination
VS	Vehicle size
PC	Parking cost
DT	Driver type
DU	Duration
ET	Estimated time of arrival to the destination
FL	Flexibility

Table 5.1: RM format of reserving a vacant parking space

- DE: The driver can enter the preferred journey destination on the OBU interface to locate the desired parking zone [39]. This will be in the form of the first 3 digits of the UK postcode e.g. LE1.
- VS: The OBU database provides details of the physical length of the vehicle (small 4 metres, medium 5 metres or large 6 metres) in line with the vehicle profile to allocate an appropriate parking space [20].
- PC: The driver can enter the preferred type of parking on the OBU interface (free or payable parking zone).
- DT: Shows the type of driver (disabled or non-disabled) based on the driver profiling retrieved from the OBU database.
- DU: The driver can enter the preferred period of stay in the parking zone (in minutes) on the OBU interface.
- ET: Utilising the GPS, this will provide the vehicle's current location and calculate the estimated time of arrival to the destination.
- FL: Depending upon the flexibility of the driver, the ISC will transmit appropriate parking options/ spaces (e.g. if the driver is flexible) to identify and transmit alternative parking zones.

5.2.2 The Master Data Repository

As mentioned earlier, the MDR in the ISC tier contains up-to-date information about all ISs policies across the parking system. The following Table 5.2 describes the ISs policies in parameters:

- ISID: Each IS across the system has a unique ID number to distinguish it from others.

Policies	Description
ISID	InfoStation unique ID number
ISL	IS location address
AS	Available space
ISPC	Parking cost
PS	Parking scheme
ISDU	Duration allowances
AT	Estimated time of arrival

Table 5.2: The MDR entities

- ISL: Each parking zone across the parking system will use the first 3 digits from the respective UK post code.
- AS: The IS measures the vacant parking space in meters (e.g. 3m, 5m etc.) to match it with the vehicle's size.
- ISPC: The IS decides whether it is a free parking zone or payable.
- PS: The IS decides whether it is a disabled on non-disabled parking zone.
- ISDU: The maximum time allowed to park in the parking zone.
- AT: The requested parking zone will only be shown as 'Available' if the estimated time of arrival (as transmitted by the vehicle) is less than 10 minutes.

5.3 Reservation Mechanism

The activity diagram in Figure 5.1 shows the entire process of the parking space reservation which is divided into three tiers, starting with the vehicle tier. This first tier represents the processes related to the vehicle combined with the OBU. The second tier represents the processes related to the IS of a parking zone. The third tier represents the processes related to the ISC which is the local traffic authority. The three tiers co-operate in unison while executing the processes related to the parking reservation system.

Once the information is received by the ISC, the validation process is initiated as shown in Figure 5.2. This stage channels the information and selects an outcome based upon the parameters as described in the next section.

5.3.1 Function 1: Validation

Once the RM is received, this is matched with the appropriate IS policy. If the RM fits the parameters, the process is directed to the next policy and so on. The first policy to be assessed is the IS location, which is recorded as the first 3 digits of the relevant UK postcode (e.g. LE1). If the vehicle's requested destination falls within the IS location, the IS moves on to enforce the next policy.

Following the recognition of the IS location, the ISC matches the vehicle's size to the space availability in the selected parking zone. The ISC uses a technique to cater for pulling-in and pulling-out of the selected parking space. This is carried out by adding an additional 2 metres (1 metre in front and 1 metre behind the selected parking space) to the vehicle's size as found in the RM. For example, a small vehicle (4 metres) will only be allocated a space of 6 metres or larger.

If successful, the IS will consider the next policy, which is parking cost. This is categorised into free and payable parking zones. If the driver requires a free parking space, only spaces which are free will be selected. If the driver is classed as payable, both free and payable parking spaces will be selected.

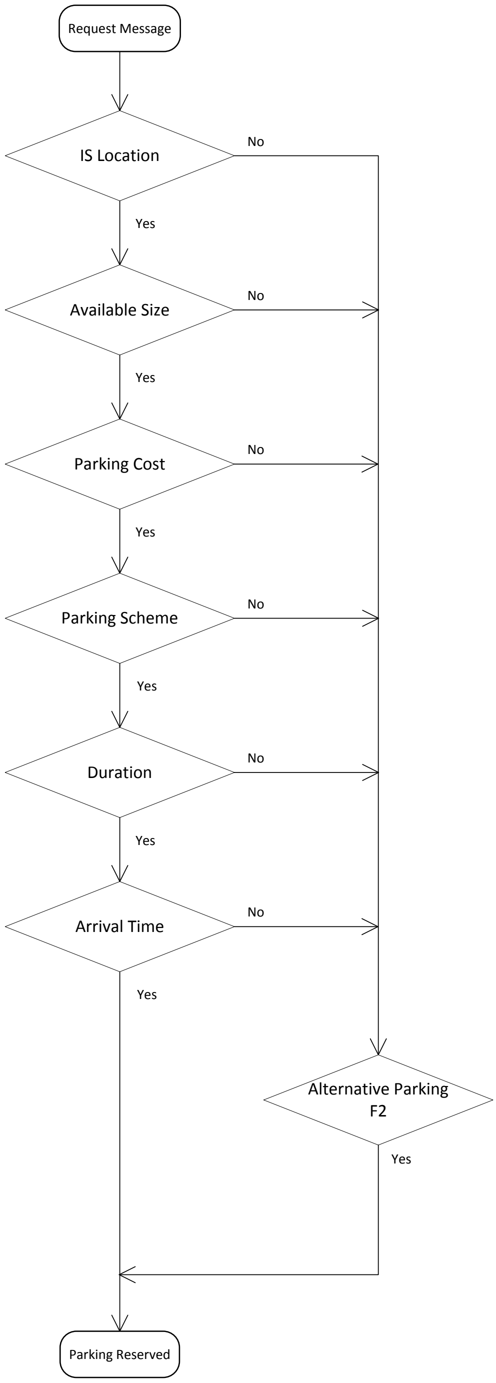


Figure 5.2: Function 1 The Validation Process

Following this, the ISC will consider the parking scheme employed. This is categorised into disabled and non-disabled parking zones. If the driver is classed as non-disabled, only non-disabled parking spaces will be selected. If the driver is classed as disabled, both disabled and non-disabled parking zones will be selected.

The next policy to be employed is duration of stay. The driver's preferred duration of stay will be considered alongside the maximum parking allowed in the IS policy. This will vary according to local authority policies in place. For example, if a parking zone has a policy which restricts parking to 1 hour without return, only drivers with a preferred duration of stay not exceeding 60 minutes will be considered for the selected parking zone.

Finally, the estimated time of arrival as calculated based on the vehicle's current location (as shown by the OBU) will be considered alongside the arrival time policy. All ISs will allow a maximum of 10 minutes arrival time for prospective vehicles. If a vehicle's expected arrival time is within 10 minutes, the available parking spaces will be selected.

If at any stage, the RM does not match the IS policies, the process will be directed to the alternative function (Function 2 in 5.3.2).

5.3.2 Function 2: Alternative

This function will only be enabled if the driver is found to be flexible according to the RM (including the driver preferences). The RM parameters are classified into two categories: fixed and non-fixed (see Table 6.10). Fixed parameters cannot be altered whereas non-fixed parameters can be altered according to driver preference (e.g. preferred destination can be expanded from LE1 to include LE2 and other surrounding IS locations). These adjusted parameters will allow for alternative parking spaces to be selected.

The first variable considered is the non-fixed parameter of the desired destination. If the preferred IS location does not contain any available parking spaces, the ISC will expand the search to the surrounding IS locations, as mentioned previously. If there are available parking spaces in the preferred IS location but the spaces are insufficient to cater for the vehicle size, the ISC will expand the search to the surrounding IS locations with available parking spaces to match the vehicle size (inclusive of 2 metres manoeuvring space), which is a fixed parameter.

The parking cost is the next one that is classed as non-fixed parameter. If the driver is classed as payable and no parking spaces are available in the preferred IS location, alternative parking in the surrounding IS locations will be identified. Both free and payable parking spaces will be identified in accordance with the flexibility of the driver being classed as payable. The alternative function considers the fixed parameter of the driver type (either as disabled or non-disabled). If no parking spaces are available in the preferred IS location, alternative parking spaces are identified in the surrounding IS locations in accordance with the respective driver type. Disabled drivers will be eligible for parking spaces in both disabled and non-disabled parking zones whereas non-disabled drivers will only be shown available spaces in non-disabled parking zones.

The non-fixed parameter of the duration of stay can be altered if no available parking spaces are found in the desired destination. In this scenario, the ISC will search for available parking spaces with a duration of stay less than the initial preferred duration of stay in accordance with the duration flexibility as designated by

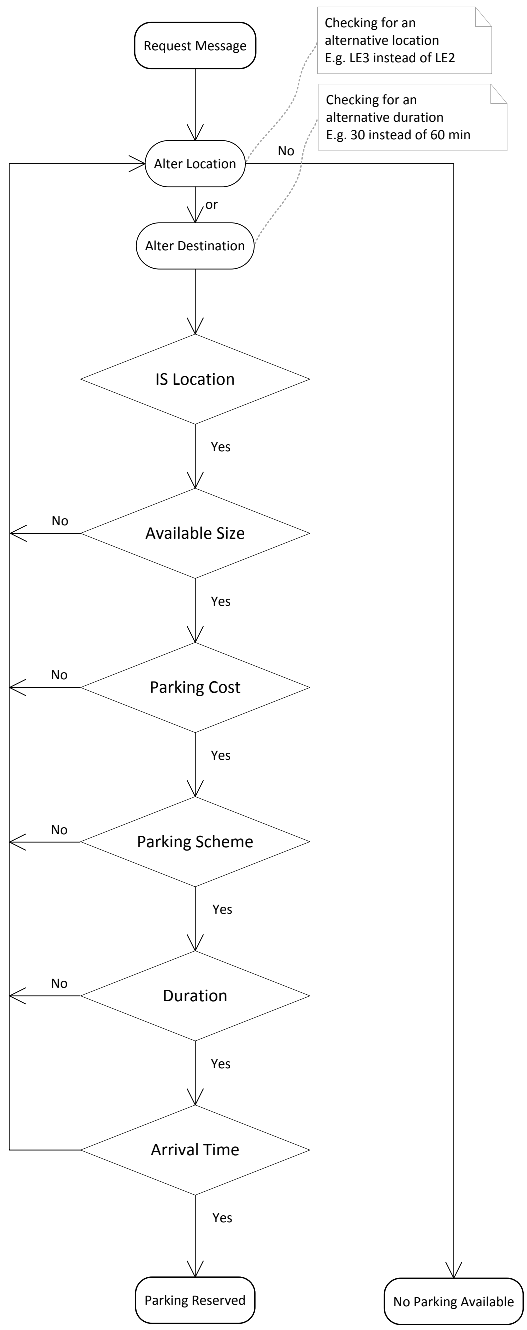


Figure 5.3: Function 2 The Alternative Process

the driver. For example, if a driver desired a parking space for 1 hour and set his duration flexibility for up to 50% of the preferred duration stay, only parking spaces with an availability of 30 minutes or more will be shown. If no parking spaces are found in the preferred IS location, the search will be expanded to the surrounding IS locations with the same duration parameters enforced.

If the expected time of arrival exceeds 10 minutes, no available parking spaces in the preferred IS location will be shown. As this is a fixed parameter, there is a limited degree of flexibility. In this case, alternative parking spaces in surrounding IS locations which are reachable within 10 minutes from the vehicle's location will be identified and selected. Once a suitable parking space has been identified, the system will reserve the parking space and send confirmation to the driver. If all the IS policies have been enforced and still no suitable parking space has been identified, then the ISC will send a notification, informing the driver that no available parking spaces have been identified.

5.4 Formalising the System in CCA

The building blocks of the system's architecture as shown in Figure 4.3 in the previous chapter, which is based on IS and context-aware consists of one centralised ISC, a number of ISs and many vehicles. This section proposes a model in which each node of the on-street parking reservation system is an ambient.

The calculus which will be used for modelling mobile systems that are context-aware will be in the CCA syntax. Based on the concept of ambient as defined in the calculus of Mobile Ambients, it provides primitives for modelling mobility and context-awareness. This formal method has been chosen to specify the work because of CCA's modularity and because it can model mobile, context-aware, concurrent systems [88].

The central node of the system is the ISC ambient which operates in parallel with the IS ambient. The IS represents a specific parking zone and provides access to the parking service as a whole. In this model, a child ambient of the IS ambient will be a vehicle that is within the range of an IS and where it can travel freely across the ISs coverage as illustrated in Figure 5.4. This section will also present the syntax of the calculus.

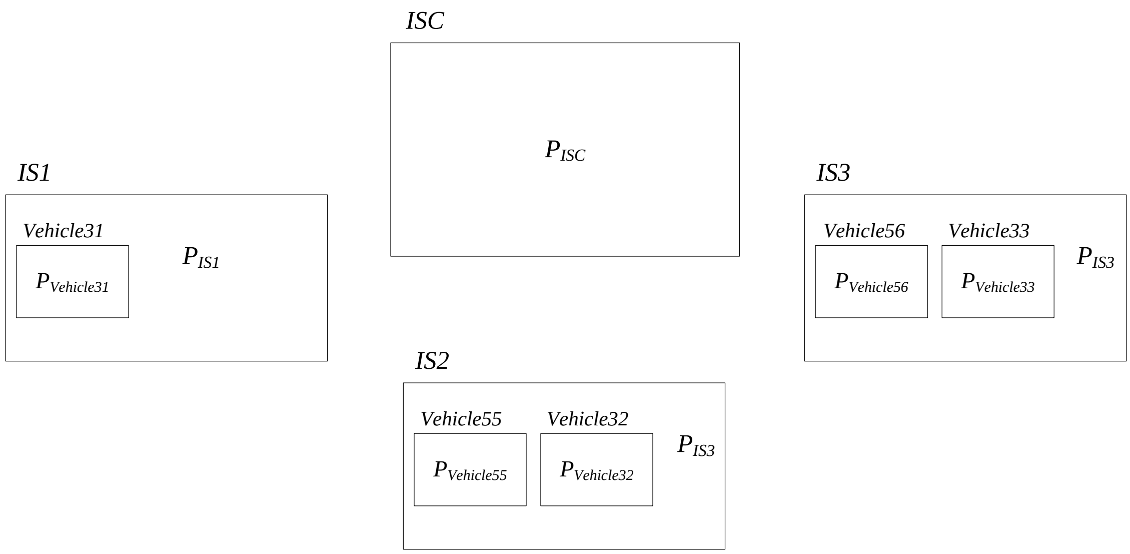


Figure 5.4: A model of the IS based on-street parking system

When a vehicle makes a request, the OBU installed in the vehicle sends the RM to the nearest IS. As illustrated in the CCA graphical representation in Figure 5.4, Vehicle 31 has moved inside the IS1 network coverage where it will be able to receive, process and forward the RM to the ISC. This graphical representation is depicted in the CCA syntax below:

$$ISC\ [P_{ISC}] \mid IS1\ [Vehicle31\ [P_{Vehicle31}] \mid P_{IS1}]$$
$$\mid IS2\ [Vehicle55\ [P_{Vehicle55}] \mid Vehicle32\ [P_{Vehicle32}] \mid P_{IS2}]$$
$$\mid IS3\ [Vehicle56\ [P_{Vehicle56}] \mid Vehicle33\ [P_{Vehicle33}] \mid P_{IS3}]$$

The syntax shows the process P_x modelling the ambients behaviour. The on-street parking system model shown in Figure 5.4 provides a high-level graphical representation that gives an overview of the system. The low level syntax notation with details of the internal workings between the ISC, ISs and $Vehicles$ nodes will be explained in the next section showing the formalisation of the system's components.

Notation	Description
ISC	One immobile ISC
IS	Many immobile ISs e.g.IS1,IS2, ...
Vehicle	Many mobile Vehicles e.g. Vehicle1,Vehicle2
Driver	Vehicle's driver
RM	Parking request message
Reserved	Parking reservation confirmed
No Parking	No match found
Alternative	Alternative parking reservation confirmed

Table 5.3: Constants used in the CCA formalisation

5.4.1 System Model

The environment in the proposed system is '$Parking$'. Each node might represent a $Vehicle$, an IS, or an ISC.

$$Parking \;\widehat{=}\; ISC[P_{ISC}] \mid IS1[V1[P_{V1}] \mid P_{IS1}] \mid \ldots \mid IS_n[V_n[P_{Vn}] \mid P_{ISn}] \quad (5.1)$$

1. The node '$Vehicle$' is modelled as an ambient on the following structure:

$$id[P_{id} \mid RM[P_{RM}]] \quad (5.2)$$

   where

   - id defines the 'identity' of the node under consideration
   - Pid is the process that represents the node's capabilities, e.g. communication between nodes and interpreting the sensor data.
   - RM stands for the parking request message ambient which include all the parameters.

   The 'Process' is an ambient that contains the process id , process details and node capabilities, e.g. how to communicate and sense other nodes.

$$P_{id}[rm[P_{rm}] \mid IS[P_{IS}] \mid ISC[P_{ISC}] \mid msgsend[P_{msgsend}] \mid msgrecv[P_{msgrecv}]] \quad (5.3)$$

2. The node 'IS' is modelled as an ambient on the following structure.

$$id[P_{id} \mid prk_{policies}[P_{prkpolicies}]] \quad (5.4)$$

   Where

   - IS represents the structure of the ambient '$parking$'
   - $prk_{policies}$ represents the parking policies which are stored and retrieved

from the MDR on the 'ISC' node.

3. The 'ISC' node starts matching the RM with the IS policies, and simultaneously updates the MDR. It can be modelled in the following way:

$$id[IS \mid IS_{rmforward}[P_{rmforward}] \mid IS_{validate}[P_{validate}] \mid$$
$$IS_{alternative}[P_{alternative}] \mid reply[P_{reply}] \mid ISC_{update}[P_{update}]] \qquad (5.5)$$

where

- $IS_{rmforward}$ represents the instance when the IS passes the RM to the ISC
- $IS_{validate}$ stands for when the ISC receives the RM from the IS and scans through all the ISs to find a parking for the vehicle
- $IS_{alternative}$ represents the moment when the ISC searches for and reserves an alternative parking space
- $reply$ marks the point when the ISC replies to the vehicle through the IS with the message as to whether it is 'Available' or 'Not Available'
- ISC_{update} symbolises the instance when the ISC updates its MDR once the parking has been reserved.

5.4.2 Context Expression

A 'context expression' is a syntax that lists the pre-requisite characteristics of the environment which are necessary for executing the process. This section illustrates the formalisation process of the ISC, IS based context-aware on-street parking reservation system, and the vehicle. The systems-behaviour is formalised around how the components interact with each other throughout, from processing the RM to reserving the parking bay. The mathematical notation of CCA will be utilised for the formalisation process.

The behaviours formalisation apposite to each type of nodes ($Vehicle$ nodes, IS nodes and the ISC node) are specified in the following sub-sections:

5.4.2.1 Vehicle Ambient

The vehicle node is the node that initiates the parking system, willing to send the RM to a parent node IS. So its capabilities are modelled by the following syntax:

$$V[in\ IS.0 | \uparrow \langle Vid, DE, VS, PC, DT, DU, TD \rangle \,.\, \uparrow (reply, dest).0] \tag{5.6}$$

When the vehicle wants to sent a RM to the parking services, first the vehicle 'V' should fall in one of the system infrastructure network converge which is the 'IS'. And to use this services the vehicle ambient will get **in** the 'IS' ambient using the mobility capability **in** where the vehicle will performs the capability **in** n moves into the parent ambient 'IS' to initiate a child to parent communication.

Now the vehicle able to send the RM to parking system to the IS. The symbol '$\uparrow$' denotes the child to parent communication. So the $Vehicle$ capable to send the seven RM parameters in an angle brackets $\langle Vid, DE, VS, PC, DT, DU, TD \rangle$ to any parents (which is the IS in this case). The RM will be processed in the 'ISC' side and then a reply message will be send back to the source vehicle. The vehicle will receive the reply message and the desired destination in a round brackets ($reply$,des) from any parent '$\uparrow$' .

The dot symbol '•' denotes the sequential composition of processes. And the '|' denotes the concurrent execution of the processes. The process '**0**' known as inactivity process, and terminates the process. The pair of the square brackets '[' and ']' outline the boundary of the ambient vehicle 'V'.

5.4.2.2 InfoStation Ambient

The 'IS' ambient interact as an intermediate node. Its receives the RM parameters including the sender *id* 'sender' and then forwards the message to ISC. So its capabilities are modelled by the following syntax:

$$\left\{ \ Eq.3.Eq.5.5 \mid ISC :: (sender, reply, dest).sender \downarrow \langle reply, dest \rangle \, .\mathbf{0} \ \right\} \tag{5.7}$$

Here the 'IS' ambient work as a gateway forwarding the RM between the 'V' ambient and the 'ISC' ambient. In fact, the parent 'IS' ambient hold the 'V' ambient as a child ambient. On the same time the 'IS' standing and communicate in parallel with the 'ISC' ambient. The process in the 'IS' start from receiving the RM parameters and the sender *id* from any child '↓' which is in this case the 'V' ambient.

$$!\downarrow(sender, DE, VS, PC, DT, DU, TD) \tag{5.8}$$

Once the 'IS' receive the RM, it will forward it straight away to the 'ISC' to process the parking request there. Here the 'IS' and the 'ISC' are sibling ambients. Where the '::' symbol denotes the sibling interacting process.

$$ISC :: \langle sender, DE, VS, PC, DT, DU, TD \rangle \tag{5.9}$$

After processing the RM in the ISC side, the 'IS' receive the reply message from the 'ISC' and then forward it to the same source vehicle 'V'.

$$!ISC :: (sender, reply, dest) \tag{5.10}$$

$$sender \downarrow \langle reply, dest \rangle \tag{5.11}$$

5.4.2.3 InfoStation Centre Ambient

The ISC ambient contains all the policies and parking spaces information controlled by that ISs. It is the ambient that stores the details of the parking space under a particular IS and helps in verifying the RM in accordance with the parking policies.

$$ISC \left[(E.q\ 5.12).(E.q\ 5.13) \left\{ (E.q\ 5.14) \left\{ \begin{array}{l} (E.q\ 5.15)(E.q\ 5.16).0 \mid \\ (E.q\ 5.17)(E.q\ 5.18).0 \mid \\ find\ (E.q\ 5.19)\ in \\ \left\{ \begin{array}{l} (E.q\ 5.20)(E.q\ 5.21).0 \mid \\ (E.q\ 5.22)(E.q\ 5.23).0 \end{array} \right\} \end{array} \right\} \right\} \right]$$

The process in the 'ISC' ambient start from receiving the RM from any sibling ambients. Which is in this case the 'IS'.

$$! :: (IS, sender, DE, VS, PC, DT, DU, TD) \tag{5.12}$$

Then the vehicle size will be altered to '$VS + 2$' where the 2 represent the pulling-in and pulling-out gaps in meter.

$$let\ vs = VS + 2 \tag{5.13}$$

Now, the 'ISC' will retrieve the ISs policies from the MDR preparing for the matching process

$$memory \mid > (IS, ISL, AS, ISPC, PS, ISDU, AT) \tag{5.14}$$

1. **Validation**

   Now, the matching and validating process start from here. The validation mechanism will matched the received RM parameters with the IS policies.

$$\left.\begin{array}{l}\langle DE = ISL \wedge vs \leq AS \wedge (ISPC = PC \vee ISPC = all) \\ \wedge (PS = all \vee PS = DT) \wedge ISDU \geq DU \wedge AT \geq ET\end{array}\right\} \tag{5.15}$$

   If the above equation satisfied, then the 'ISC' will send back a parking confirmation message and allocated location and duration to the 'IS'.

$$IS :: \langle sender, \texttt{PARKING\_ RESERVED\_IN}, DE, IS, DU\rangle \tag{5.16}$$

   In fact, the 'ISC' will go through the MDR records until it is find the matched policies. In case there is no record matching with the RM:

$$\left.\begin{array}{l} <\neg(DE = ISL \wedge vs \leq AS \wedge (ISPC = PC \vee ISPC = all) \\ \wedge(PS = all \vee PS = DT) \wedge ISDU \geq DU \wedge AT \geq ET) > \end{array}\right\} \tag{5.17}$$

Where the symbol $<\neg$ denoted to negation. So in case if there is no matching at all, the 'ISC' will send back a message where there is no parking available satisfying the RM.

$$IS :: \langle sender, \texttt{No\_PARKING\_AVAILABLE\_IN}, DE, IS, DU\rangle \tag{5.18}$$

2. **Alternative**

As described previously in 5.3.2, the alternative function allowd to alter some of the RM parameters values. In fact, this function can be processed only when the flexibility flag 'FL' assigned to 'yes'.

$$\texttt{find de} : (\diamondsuit(\texttt{at2}(\texttt{DE},\texttt{de})) \wedge \texttt{FL} = \texttt{yes}) \texttt{ for let du} = \texttt{DU}/2 \texttt{ in} \tag{5.19}$$

After altering the parameters, the matching and validating process start from here. The validation mechanism will matched the new altered RM parameters with the policies that are retrieved from the MDR.

$$\left.\begin{array}{l} <(DE = ISL \vee de = ISL) \wedge vs <= AS \wedge (ISPC = PC \vee \\ ISPC = all) \wedge (PS = all \vee PS = DT) \wedge (ISDU >= \\ DU \vee ISDU >= du) \wedge DU < ISDU \wedge AT >= ET > \end{array}\right\} \tag{5.20}$$

If the above equation satisfied, then the 'ISC' will be send back the alternative parking confirmation message and allocated alternative location and duration to the 'IS'.

$$IS :: \langle sender, \texttt{ALTERNATIVE\_PARKING\_RESERVED\_IN}, ISL, IS, DU \rangle \qquad (5.21)$$

In fact, the 'ISC' will go through the MDR records until it is find the matched pollicises. In case there is no record matching with the new altered RM:

$$\left.\begin{array}{l} < \neg((DE = ISL \vee de = ISL) \wedge vs <= AS \wedge (ISPC = PC \vee \\ ISPC = all) \wedge (PS = all \vee PS = DT) \wedge (ISDU >= \\ DU \vee ISDU >= du) \wedge AT >= ET) > \end{array}\right\} \qquad (5.22)$$

So in case if there is no matching at all, the 'ISC' will send back a message where there is no parking available satisfying the new altered RM to the 'IS'.

$$\langle sender, \texttt{NO\_ALTERNATIVE\_PARKING\_IN}, DE, IS, DU \rangle \qquad (5.23)$$

Where this is the last message to confirms that there is no parking pays available even after altering the values of the RM parameters.

5.5 Summary

This chapter built on the novel intelligent parking reservation architecture for managing the parking space location and reservation using the concept of VANET and context-aware as a step towards improving the ITS and making the parking reservation system more user-friendly and comfortable for drivers which was presented in the previous chapter.

Here, a parking space reservation activity diagram shows the work flow between various components of the system from the instant the driver initiates the parking request through the OBU until the system allocates and reserves the proper parking space.

The proposed parking reservation architecture has been formalised in this chapter using the CCA notation in order to prove and validate the parking reservation mechanism. The Chapter went on to describe the functions performed by the various nodes depending on matches/mismatch between RM contents and the parking policies. Proposals have also been made in relation to the context-expressions in CCA which will be useful for implementing the system's mechanism.

In the following chapter, the execution environment of the CCA, namely the Calculus of Context-aware Ambients Programming Language (ccaPL), and the CCA execution processes of the IS based context-aware on-street parking system will be presented.

Chapter 6

Behaviour Evaluation

Objectives:

- To introduce the ccaPL programming language
- To describe the parking space reservation experiments
- To demonstrate the usage of the ccaPL
- To implement the experiments scenarios.
- To conclude the verification of the experiments

6.1 Introduction

This chapter intends to evaluate some properties regarding the on-street parking reservation system. The formalised system specifications outlined in Chapter 5 will be used here and will run in parallel with all the processes of the ISC and the IS's with examples of the request-parking behaviour scenarios. In the first instance however, the system's CCA specification will be converted to the Calculus of Context-aware Ambients Programming Language (ccaPL). The CCA interpreter will be used to compile the ccaPL syntax.

From here on, the behaviours in the forthcoming scenarios will be recorded using the ccaPL syntax. Some properties will be validated particularly for the on-street parking reservation system. The validation process will consist of analysing three experiments, each with two scenarios. All in all, the scenarios will test the efficiency of the system by presenting various hypothetical parking situations to the system. The system's response will be noted in each example.

It is anticipated that the tests will reveal whether the system is working correctly or not in accordance with the CCA model and the formalised RM and IS policies of the context-aware based system.

6.2 The Calculus of Context-Aware Programming Language

The proposed context-aware based system and the mobile ad hoc capability are both compatible with the CCA mathematical specifications as asserted in Chapter 5. The programming language for the CCA, known as ccaPL is an executable code of the converted CCA syntax [88]. The ccaPL interpreter is based on the Java platform which can run on any operating system that can support the Java Virtual Machine (JVM) [38]. This chapter shows how all the experiments will be executed using the ccaPL interpreter.

6.3 Experiment Trials and Findings

The experiments are divided into three possible paradigms: experiment one tests the system for valid parking in accordance with Function 1 in section 5.3.1 which is the first stage of the validation process; experiment two tests the system for non-valid parking in accordance with Function 1 in section 5.3.1 which is the first stage of the validation process; and experiment three tests the system for alternative valid parking which utilises the context-aware ability in accordance with Function 2 in section 5.3.2 which is the last stage of the alternative validation process.

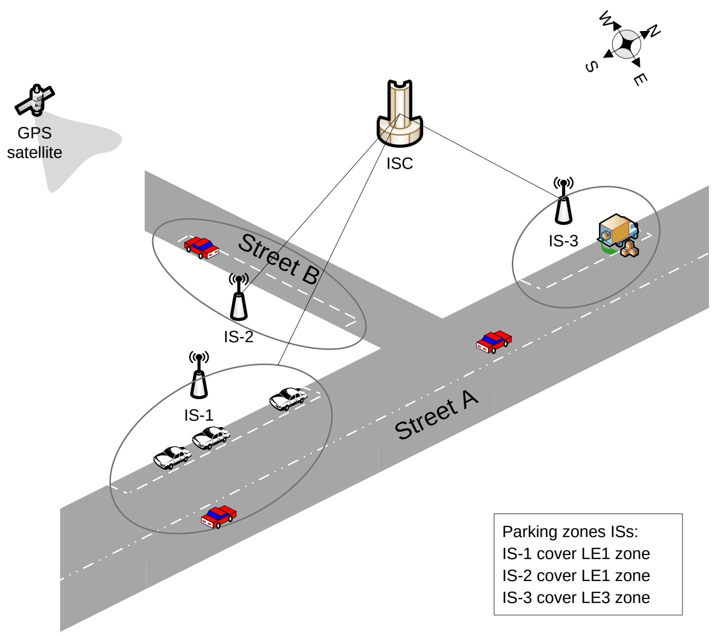


Figure 6.1: Possible path for the Requesting Message

In all the experiments, the immobile ISC is the centralised system which represents the local traffic authority that controls the parking systems city-wide and the immobile ISs acts as a gateway between the ISC and the mobile vehicles. It will be assumed that VANET will provide coverage of the on-street parking reservation system. It should be noted here that for the sake of simplicity only the first three digits of any postcode will be utilised in the reservation mechanism. It is a noteworthy reminder here that within all these structures and processes, the RM and MDR which have been described in Chapter 5 are integral to the system.

The deployment of the proposed system nodes, namely the vehicles, ISC and IS's is based on the illustration in Figure 6.1 and shows the possible path for the RM. The parking situations in the forthcoming scenarios emerge from this model location. The experiments IS policies will be based on the proposed data provided in Table 6.1.

ISID	ISL	AS	ISPC	PS	ISDU	AT
$IS-1$	LE1	6	Payable	All	60	10
$IS-2$	LE1	7	Payable	All	90	10
$IS-3$	LE3	8	Free	Disabled	120	10
$IS-44$	LE4	12	Free	All	60	10
$IS-56$	LE5	9	Free	All	60	10
$IS-67$	LE6	0	Payable	All	120	10
$IS-77$	LE7	4	Payable	Disable	60	10
$IS-80$	LE8	10	Free	All	180	10
$IS-91$	LE9	0	Free	All	120	10

Table 6.1: An examples of the IS's policies stored on the MDR

6.3.1 Experiment 1: Valid Parking

In this study, the parking space reservation search result confirms there to be a valid parking space based on the content of the RM as assumed in the following scenarios. There is one ISC, three ISs covering three different parking zones, and two vehicles travelling in the IS coverage range as shown in Figure 6.2. Each parking zone has its own pre-set policies which are stored in the MDR as shown in Table 6.1. Two different parking-reservation behaviours will be presented in the following two scenarios. Both scenarios show the validity of the system in finding the exact parking space requested.

Figure 6.2 shows two separate parking zones in LE1: the zone based on Street A (South) is monitored by IS1 and the zone based at Street B (West) is monitored by IS2. The parking zone in LE3 on Street A (North) is monitored by IS3. All three ISs are connected to the ISC to synchronise and update the parking information.

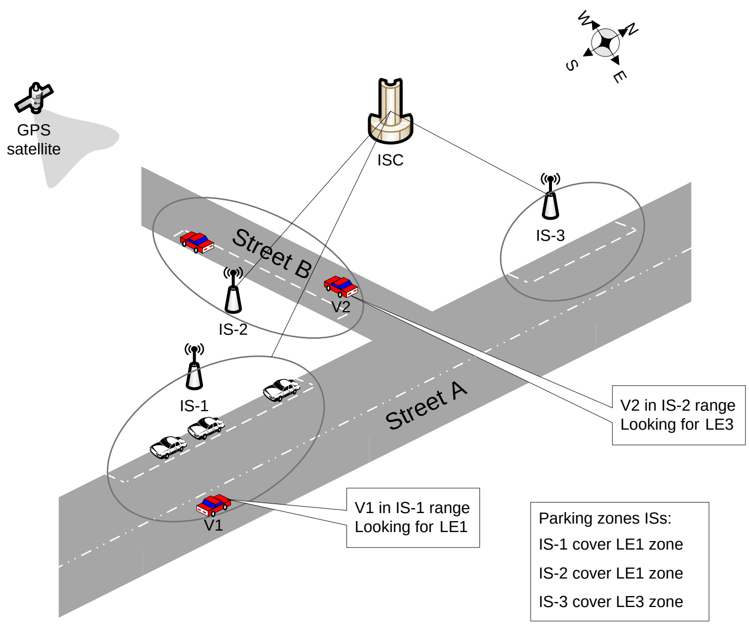


Figure 6.2: Parking request scenarios in the valid parking experiment

6.3.1.1 Valid Parking Scenario1

In Figure 6.2, vehicle V1 sends a parking request while it is in the coverage range of IS1. It looks for a parking space in LE1 where there are two parking zones which match with the contents of the RM in Table 6.2. The IS1 receives the RM and then forwards it to the ISC to start processing and matching it with the MDR which has been referred to in Table 6.1. In response, the ISC starts to query the MDR.

DE	VS	PC	DT	DU	ET
LE1	Small	Payable	Non-disabled	60 min	1 min

Table 6.2: Contents of the RM for valid parking Scenario 1

In this scenario, the driver who is driving a small size vehicle is only interested in the LE1 area. The driver makes a request for a payable parking zone, which means the system will also allow the driver to use the free of charge parking bay. The driver is a non-disabled driver so will therefore only be allowed to use the non-disabled parking bays. Finally, the OBU in the vehicle is expecting to arrive at the destination within 1 minute based on the current location of the vehicle. The ISC accepts and allocates a parking bay in parking zone LE1 which is within the coverage range of IS1 by meeting the preferred parameters of the RM.

The execution output of the ccaPL for this scenario is shown in Table 6.3:

```
1.--->  {call to the abstraction "root@memory" in the ambient "ISC"}
2.--->  {ambient "V1" moves into ambient "IS1"}
3.--->  {Child to parent: V1 ===(V1,LE1,4,payable,non_disabled,60,1)===> IS1}
4.--->  {Sibling to sibling: IS1 ===(IS1,V1,LE1,4,payable,non_disabled,60,1)===> ISC}
5.--->  {Parent to child: ISC ===()===> IS1}
6.--->  {Local: IS1 ===(LE1,6,payable,all,60,10)===> IS1}
7.--->  {Child to parent: IS1 ===(LE1,6,payable,all,60,10)===> ISC}
8.--->  {Sibling to sibling: ISC ===(V1,Parking_Reserved_in,LE1,IS1)===> IS1}
9.--->  {Parent to child: IS1 ===(Parking_Reserved_in,LE1,IS1)===> V1}
```

Table 6.3: The ccaPL output of valid parking Scenario 1

The syntax in the ccaPL output in Table 6.3 can described in the following way. the syntax '`--->`' stands for the reduction relation to the CCA which verifies a process transition as properly described in [88]. the description of every transition is given between curly brackets. In particular, the notation '`A ==> (x,y,z) ==> B`' means that an ambient 'A' has sent a message with contents '`x,y,z`' to another ambient 'B'. other notations such as 'child to parent' and 'sibling to sibling' provide information about the relationship between the sender 'A' and the receiver 'B' [38].

The explanation of the output's syntax in the valid parking scenario 1 is as follows. The ISC starts by calling the database records. The ambient V1 then acts on the capability and moves into the sibling ambient IS1 to become a child of the parent IS1 and V1 starts sending the RM to IS1. The IS1 forwards it to the ISC (sibling to sibling) in order to process the RM and sends the reservation details back to source V1. Line 6 shows the matching process between the RM and the IS1 policies. This results in the final response message which confirms the parking reservation and location as requested in the original RM.

The complete ccaPL code for this scenario can be found in Appendix A.

6.3.1.2 Valid Parking Scenario 2

In Figure 6.2, vehicle V2 sends a parking request while it is in the coverage range of IS2. It looks for a parking space in LE3 where there is only one parking zone which matches with the contents of the RM in Table 6.4. The IS2 receives the RM and then forwards it to the ISC to start processing and matching it with the MDR which has been referred to in Table 6.1. In response, the ISC starts to query the MDR.

DE	VS	PC	DT	DU	ET
LE3	Large	Free	Disabled	60 min	10 min

Table 6.4: Contents of the RM for valid parking Scenario 2

In this scenario, the driver who is driving a large size vehicle is only interested in the LE3 area. The driver makes a request for a free of charge parking zone, which means the system will allow the driver to use the free of charge parking bay only. The driver is a disabled driver so will therefore be allowed to use the non-disabled parking bays also. Finally, the OBU in the vehicle is expecting to arrive at the destination within 10 minutes based on the current location of the vehicle. The ISC accepts and allocates a parking bay in parking zone LE3 which is within the coverage range of IS3 by meeting the preferred parameters of the RM.

The execution output of the ccaPL for this scenario is shown in Table 6.5:

```
1.--->  {call to the abstraction "root@memory" in the ambient "ISC"}
2.--->  {ambient "V2" moves into ambient "IS2"}
3.--->  {Child to parent: V2 ===(V2,LE3,6,free,disabled,60,10)===> IS2}
4.--->  {Sibling to sibling: IS2 ===(IS2,V2,LE3,6,free,disabled,60,10)===> ISC}
5.--->  {Parent to child: ISC ===()===> IS1}
6.--->  {Local: IS1 ===(LE1,6,payable,all,60,10)===> IS1}
7.--->  {Child to parent: IS1 ===(LE1,6,payable,all,60,10)===> ISC}
8.--->  {Parent to child: ISC ===()===> IS2}
9.--->  {Local: IS2 ===(LE1,7,payable,all,90,10)===> IS2}
10.--->  {Child to parent: IS2 ===(LE1,7,payable,all,90,10)===> ISC}
11.--->  {Parent to child: ISC ===()===> IS3}
12.--->  {Local: IS3 ===(LE3,8,all,disabled,120,10)===> IS3}
13.--->  {Child to parent: IS3 ===(LE3,8,all,disabled,120,10)===> ISC}
14.--->  {Sibling to sibling: ISC ===(V2,Parking_Reserved_in,LE3,IS3)===> IS2}
```

Table 6.5: The ccaPL output of valid parking Scenario 2

The explanation of the output's syntax in the valid parking scenario 2 is as follows. The ISC starts by calling the database records. The ambient V2 then acts on the capability and moves into the sibling ambient IS2 to become a child of the parent IS2 and V2 starts sending the RM to IS2.

The IS2 forwards it to the ISC (sibling to sibling) in order to process the RM and sends the reservation details back to source V2. Line 9 shows the matching process between the RM and the IS2 policies. This results in the final response message which confirms the parking reservation and location as requested in the original RM.

The complete ccaPL code for this scenario can be found in Appendix A.

6.3.2 Experiment 2: Non-Valid Parking

In this study, the parking space reservation search result confirms there to be a non-valid parking space based on the content of the RM as assumed in the following scenarios. There is an ISC, two ISs covering two different parking zones, and two vehicles travelling in the IS coverage range as shown in Figure 6.3.

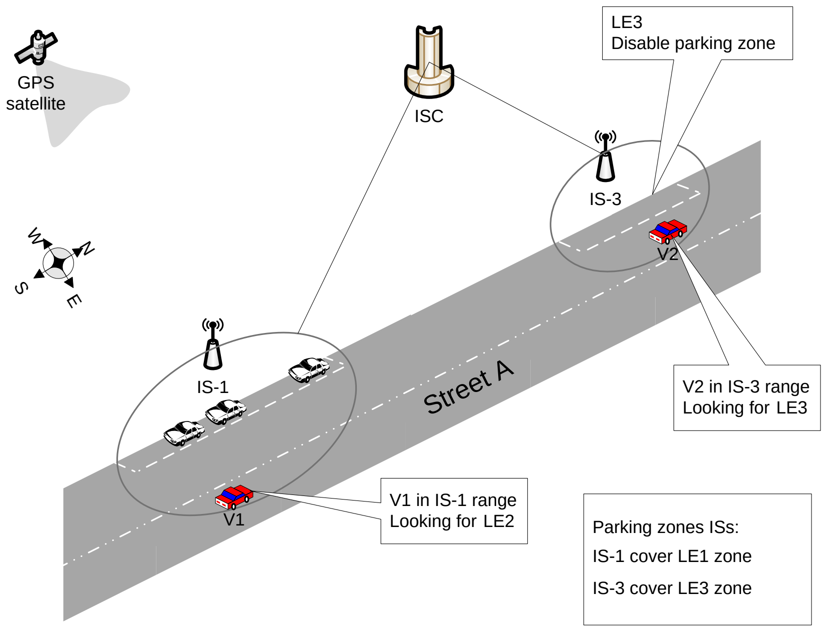


Figure 6.3: Parking requests scenarios in the non-valid parking experiment

Each parking zone has its own pre-set policies which are stored in the MDR as shown in Table 6.1. It should be noted that in this particular experiment, there is neither a parking space available nor is there an alternative.

Two different parking-reservation behaviours will be presented in the following two scenarios. Both scenarios show that the exact parking space requested cannot be matched.

Figure 6.3 shows two separate parking zones: the parking zone in LE1 based on Street A (South) is monitored by IS1 and the parking zone in LE3 on Street A (North) is monitored by IS3. The two ISs are connected to the ISC to synchronise and update the parking information. In this experiment, LE2 does not have any parking zones whatsoever and LE3 is reserved for disabled-parking only.

6.3.2.1 Non-Valid Parking Scenario 1

In Figure 6.3, vehicle V1 sends a parking request while it is in the coverage range of IS1. It looks for a parking space in LE2 where there are no parking zones at all. The IS1 receives the RM in Table 6.6 and then forwards it to the ISC to start processing and matching it with the MDR which has been referred to in Table 6.1. In response, the ISC starts to query the MDR.

DE	VS	PC	DT	DU	ET
LE2	Small	Free	Non-disabled	60 min	10min

Table 6.6: Contents of the RM for non-valid parking scenario 1

In this scenario, the driver who is driving a small size vehicle is only interested in the LE2 area. The driver makes a request for a parking bay, and although the ISC receives the RM it is unable to respond as LE2 is a parking-free zone. The ISC rejects the RM and does not allocate a parking bay in parking zone LE2. Consequently, the system responds by replying to the driver 'No Parking Available in LE2'. The ISC does not provide an alternative option.

The execution output of the ccaPL for this scenario is shown in Table 6.7:

```
1.--->  {call to the abstraction "root@memory" in the ambient "ISC"}
2.--->  {ambient "V1" moves into ambient "IS1"}
3.--->  {Child to parent: V1 ===(V1,LE2,4,free,non_disabled,60,10)===> IS1}
4.--->  {Sibling to sibling: IS1 ===(IS1,V1,LE2,4,free,non_disabled,60,10)===> ISC}
5.--->  {Parent to child: ISC ===()===> IS1}
6.--->  {Local: IS1 ===(LE1,6,payable,all,60,10)===> IS1}
7.--->  {Child to parent: IS1 ===(LE1,6,payable,all,60,10)===> ISC}
8.--->  {Parent to child: ISC ===()===> IS2}
9.--->  {Local: IS2 ===(LE1,7,payable,all,90,10)===> IS2}
10.--->  {Child to parent: IS2 ===(LE1,7,payable,all,90,10)===> ISC}
11.--->  {Parent to child: ISC ===()===> IS3}
12.--->  {Local: IS3 ===(LE3,8,all,disabled,120,10)===> IS3}
13.--->  {Child to parent: IS3 ===(LE3,8,all,disabled,120,10)===> ISC}
14.--->  {Sibling to sibling: ISC ===(V1,No_Parking_Available_in,LE2,Area)===> IS1}
15.--->  {Parent to child: IS1 ===(No_Parking_Available_in,LE2,Area)===> V1}
```

Table 6.7: The ccaPL output of non-valid parking scenario 1

The explanation of the output's syntax in the non-valid parking scenario 1 is as follows. The ISC starts by calling the database records. The ambient V1 then acts on the capability and moves into the sibling ambient IS1 to become a child of the parent IS1 and V1 starts sending the RM to IS1. The IS1 forwards it to the ISC (sibling to sibling) in order to process the RM and sends the reservation details back to source V1.

Lines 6 to 12 show the matching process between the RM and the IS's policies. This results in the final response message which confirms that there is no parking reservation available as requested in the original RM.

The complete ccaPL code for this scenario can be found in Appendix A.

6.3.2.2 Non-Valid Parking Scenario 2

In Figure 6.3, vehicle V2 sends a parking request while it is in the coverage range of IS3. It looks for a parking space in LE3 where there are no parking zones which match with the contents of the RM in Table 6.8. The IS3 receives the RM and then forwards it to the ISC to start processing and matching it with the MDR which has been referred to in Table 6.1. In response, the ISC starts to query the MDR.

DE	VS	PC	DT	DU	ET
LE3	Mid	Free	Non-disabled	60 min	1 min

Table 6.8: Contents of the RM for non-valid parking scenario2

In this scenario, the driver who is driving a medium size vehicle is only interested in the LE3 which is an area designated for disabled drivers only. So due to the fact that the driver is non-disabled, the system does not allow the driver to use a parking bay even if a space is available in LE3. Consequently, the system delivers the message 'No Parking Available in LE3' to the driver. The ISC does not provide an alternative option.

The execution output of the ccaPL for this scenario is shown in Table 6.9:

```
1.--->  {call to the abstraction "root@memory" in the ambient "ISC"}
2.--->  {ambient "V2" moves into ambient "IS3"}
3.--->  {Child to parent: V2 ===(V2,LE3,5,free,non_disabled,60,1)===> IS3}
4.--->  {Sibling to sibling: IS3 ===(IS3,V2,LE3,5,free,non_disabled,60,1)===> ISC}
5.--->  {Parent to child: ISC ===()===> IS1}
6.--->  {Local: IS1 ===(LE1,6,payable,all,60,10)===> IS1}
7.--->  {Child to parent: IS1 ===(LE1,6,payable,all,60,10)===> ISC}
8.--->  {Parent to child: ISC ===()===> IS2}
9.--->  {Local: IS2 ===(LE1,7,payable,all,90,10)===> IS2}
10.--->  {Child to parent: IS2 ===(LE1,7,payable,all,90,10)===> ISC}
11.--->  {Parent to child: ISC ===()===> IS3}
12.--->  {Local: IS3 ===(LE3,8,all,disabled,120,10)===> IS3}
13.--->  {Child to parent: IS3 ===(LE3,8,all,disabled,120,10)===> ISC}
14.--->  {Sibling to sibling: ISC ===(V2,No_Parking_Available_in,LE3,IS3)===> IS3}
15.--->  {Parent to child: IS3 ===(No_Parking_Available_in,LE3,IS3)===> V2}
```

Table 6.9: The ccaPL output of non-valid parking scenario 2

The explanation of the output's syntax in the non-valid parking scenario 2 is as follows. The ISC starts by calling the database records. The ambient V2 then acts on the capability and moves into the sibling ambient IS2 to become a child of the parent IS2 and V2 starts sending the RM to IS2.

The IS2 forwards it to the ISC (sibling to sibling) in order to process the RM and sends the reservation details back to source V2. Lines 6 to 12 show the matching process between the RM and the IS's policies. This results in the final response message which confirms that there is no parking reservation available as requested in the original RM.

The complete ccaPL code for this scenario can be found in Appendix A.

6.3.3 Experiment 3: Alternative Valid Parking

This experiment shows that in the event where the ISC cannot find the exact requested parking bay, the system will adopt the alternative function (Function 2 in 5.3.2) according to the driver preferences which are pre-set on the OBU. The aim of this mechanism is to help find the most appropriate alternative parking bay in alternative parking zones which match the RM as far as possible.

The RM contains a number of parameters that could fall into two classification: fixed and non-fixed (as shown in Table 6.10). Fixed parameters are those that cannot be modified by the driver namely: the vehicle size, the driver type and the expected time of arrival at the destination. On the other hand, non- fixed parameters are those which play a vital role during the alternative function and which give the driver the flexibility in choosing the parking. In fact, these reflect the driver's preferences.

RM	Fixedly	Flexibility	Alternative options
DE	Non-fixed	Increase/decrease search diameter	Alternative DE
VS	Fixed	Vehicle's physical size cannot be altered	Alternative DE
PC	Non-fixed	payable can use both free and payable	Alternative DE
DT	Fixed	Disabled/non-disabled cannot be altered	Alternative DE
DU	Non-fixed	Duration time can be reduce	e.g. 30 min instead of 60
ET	Fixed	Expected arrival time cannot be altered	Alternative DE

Table 6.10: The Fixed and Non-fixed parameters on the RM

In this experiment, a new parameter is being introduced, 'FL' denoting that the driver is 'flexible' in relation to the non-fixed parameters in the RM. The flexibility is with regards to the 'Destination' which consequently increases the diameter of the search, and the 'Duration' which reduces the stay by up to half of the requested time.

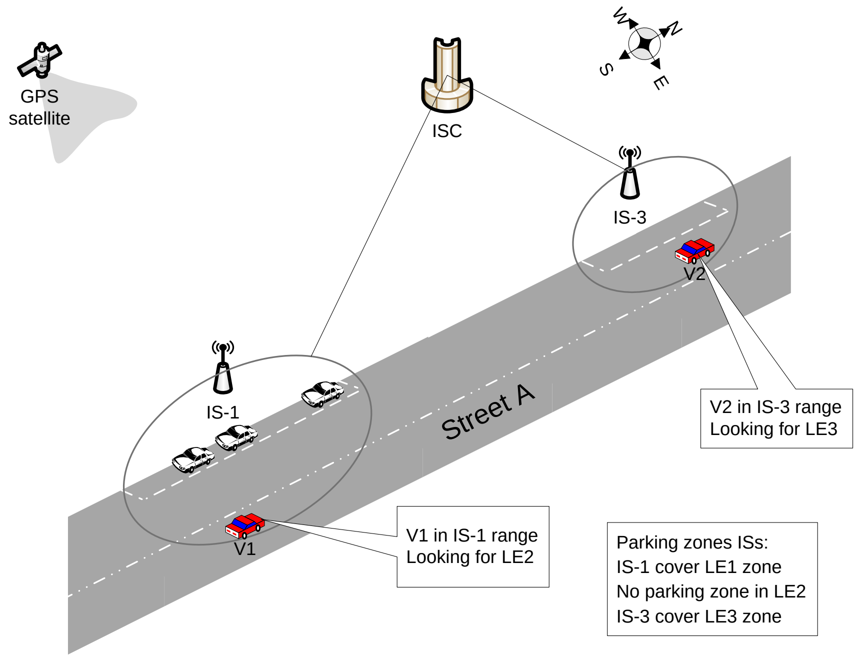


Figure 6.4: Parking requests scenarios in the alternative valid parking experiment

Figure 6.4 shows two separate parking zones: the parking zone in LE1is based on Street A (South) and monitored by IS1 and the parking zone in LE3 on Street A (North) monitored by IS3. The two ISs are connected to the ISC to synchronise and update the parking information.

6.3.3.1 Alternative Valid Parking Scenario 1

In Figure 6.4, vehicle V1 sends a parking request while it is in the coverage range of IS1. It looks for a parking space in LE2 where there are no parking zones which match with the contents of the RM in Table 6.11. The IS1 receives the RM and then forwards it to the ISC to start processing and matching it with the MDR which has been referred to in Table 6.1. In response, the ISC starts to query the MDR.

DE	VS	PC	DT	DU	ET	FL
LE2 $\rightarrow LE3$	Small	Free	Disabled	60 min	10 min	Yes

Table 6.11: Contents of the RM for the alternative valid parking scenario1

In this scenario, the driver who is driving a small size vehicle is interested in the LE2 area. The driver makes a request for a free of charge parking bay, which means the system allows the driver to use the free of charge parking bay only. The driver is a disabled driver so will also be allowed to use the non-disabled parking bays. The OBU in the vehicle is expecting to arrive at the destination within 10 minutes based on the current location of the vehicle. Finally, the driver indicates a 'flexibility' preference in the RM by affirming a 'Yes' which means the parking system allows to expand the search by altering the RM parameters.

Here, the ISC can not find a parking space in the LE2 area that match all the drivers preferences in the RM. Since the ISC adapts the alternative function in this experiment (function2), the system will check whether or not there are any flexibile preferences included by the driver in order to expand the sereach. Consequently, based on the drivers flexibilities preferences shown in Table 6.11 as FL, the system allows itself to alter matched parameter in the RM from LE2 to LE3 (an increase in the diameter search) the system then query the MDR again which will result in allocating a parking bay that match the new altered parking RM.

The execution output of the ccaPL for this scenario is shown in Table 6.12:

```
1.--->  {call to the abstraction "root@memory" in the ambient "ISC"}
2.--->  {ambient "V1" moves into ambient "IS1"}
3.--->  {Child to parent: V1 ===(V1,LE2,4,free,disabled,60,10,yes)===> IS1}
4.--->  {Sibling to sibling: IS1 ===(IS1,V1,LE2,4,free,disabled,60,10,yes)===> ISC}
5.--->  {Parent to child: ISC ===()===> IS1}
6.--->  {Local: IS1 ===(LE1,6,payable,all,60,10)===> IS1}
7.--->  {Child to parent: IS1 ===(LE1,6,payable,all,60,10)===> ISC}
8.--->  {Parent to child: ISC ===()===> IS2}
9.--->  {Local: IS2 ===(LE1,7,payable,all,90,10)===> IS2}
10.--->  {Child to parent: IS2 ===(LE1,7,payable,all,90,10)===> ISC}
11.--->  {Parent to child: ISC ===()===> IS3}
12.--->  {Local: IS3 ===(LE3,8,all,disabled,120,10)===> IS3}
13.--->  {Child to parent: IS3 ===(LE3,8,all,disabled,120,10)===> ISC}
14.--->  {Sibling to sibling: ISC ===(V1,NO_PARKING_AVAILABLE_IN,LE2,IS1,60)===> IS1}
15.--->  {Parent to child: IS1 ===(NO_PARKING_AVAILABLE_IN,LE2,IS1,60)===> V1}
16.--->  {Parent to child: ISC ===()===> IS1}
17.--->  {Local: IS1 ===(LE1,6,payable,all,60,10)===> IS1}
18.--->  {Child to parent: IS1 ===(LE1,6,payable,all,60,10)===> ISC}
19.--->  {binding: de -> LE3}
20.--->  {Parent to child: ISC ===()===> IS2}
21.--->  {Local: IS2 ===(LE1,7,payable,all,90,10)===> IS2}
22.--->  {Child to parent: IS2 ===(LE1,7,payable,all,90,10)===> ISC}
23.--->  {Parent to child: ISC ===()===> IS3}
24.--->  {Local: IS3 ===(LE3,8,all,disabled,120,10)===> IS3}
25.--->  {Child to parent: IS3 ===(LE3,8,all,disabled,120,10)===> ISC}
26.--->  {Sibling to sibling: ISC ===(V1,ALTERNATIVE_PARKING_RESERVED_IN,LE3,IS3,60)===> IS1}
```

Table 6.12: The ccaPL output of alternative valid parking Scenario 1

The explanation of the output's syntax in the alternative valid parking scenario 1 is as follows. The ISC starts by calling the database records. The ambient V1 then acts on the capability and moves into the sibling ambient IS1 to become a child of the parent IS1 and V1 starts sending the RM to IS1.

The IS1 forwards it to the ISC (sibling to sibling) in order to process the RM and sends the reservation details back to source V1. Lines 6 to 12 show the matching process between the RM and the IS's policies. This results in the first response which confirms that there is no parking reservation available as requested in the original RM. This leads to the ISC altering the 'destination' parameter from LE2 to LE3 as shown in Line 19. Line 24 shows the new matching process between the RM and the IS's policies. Line 26 confirms the reservation for the alternative parking.

The complete ccaPL code for this scenario can be found in Appendix A.

6.3.3.2 Alternative Valid Parking Scenario 2

In Figure 6.4, vehicle V2 sends a parking request while it is in the coverage range of IS3. It looks for a parking space in LE3 where there is a single parking zone which could match the contents of the RM in Table 6.13. The IS3 receives the RM and then forwards it to the ISC to start processing and matching it with the MDR which has been referred to in Table 6.1. In response, the ISC starts to query the MDR.

DE	VS	PC	DT	DU	ET	FL
LE3	large	Free	Disabled	180 $\rightarrow$ 120min	10 min	Yes

Table 6.13: Contents of the RM for the alternative valid parking scenario 2

In this scenario, the driver who is driving a large size vehicle is interested in the LE3 area. The driver makes a request for a free of charge parking bay, which means the system will allow the driver to use the free of charge parking bay only. The driver is a disabled driver so will also be allowed to use the non-disabled parking bays. The driver makes a request for the duration of 180 minutes. The OBU in the vehicle is expecting to arrive at the destination within 10 minutes based on the current location of the vehicle. Finally, the driver indicates a 'flexibility' preference in the RM by affirming a ‘Yes’ which means the parking system allows to expand the search by altering the RM parameters.

However, ISC cannot find a parking space in the LE3 area which match all the driver’s preferences in the RM. Since the ISC adopts the alternative function in this experiment (switch to Function 2), the system will check whether or not there are any ‘flexible’ preferences included by the driver in order to expand the search. Consequently, based on the driver’s ‘flexibility’ preferences shown in Table 6.13 as FL, the system allows itself to alter the miss matched parameter. ISC will alter the parking duration parameter in the RM from 180 minutes to a minimum of half the requested time, and no less. The system will then query the MDR again which will result in allocating a parking bay that matches the new altered parking RM.

The ISC will identify a parking bay which has the nearest amount of time available to the duration requested by the driver in the original RM, but the parking bay which is allocated will never have less than half the time available that was requested by the driver. So, in this scenario, the ISC will identify and allocate a disabled parking bay that has a duration anywhere between 90 minutes and 180 minutes available.

The execution output of the ccaPL for this scenario is shown in Table 6.14 :

```
1.--->  {call to the abstraction "root@memory" in the ambient "ISC"}
2.--->  {ambient "V2" moves into ambient "IS3"}
3.--->  {Child to parent: V2 ===(V2,LE3,6,free,disabled,180,10,yes)===> IS3}
4.--->  {Sibling to sibling: IS3 ===(IS3,V2,LE3,6,free,disabled,180,10,yes)===> ISC}
5.--->  {Parent to child: ISC ===()===> IS1}
6.--->  {Local: IS1 ===(LE1,6,payable,all,60,10)===> IS1}
7.--->  {Child to parent: IS1 ===(LE1,6,payable,all,60,10)===> ISC}
8.--->  {Parent to child: ISC ===()===> IS2}
9.--->  {Local: IS2 ===(LE1,7,payable,all,90,10)===> IS2}
10.--->  {Child to parent: IS2 ===(LE1,7,payable,all,90,10)===> ISC}
11.--->  {Parent to child: ISC ===()===> IS3}
12.--->  {Local: IS3 ===(LE3,8,all,disabled,120,10)===> IS3}
13.--->  {Child to parent: IS3 ===(LE3,8,all,disabled,120,10)===> ISC}
14.--->  {Sibling to sibling: ISC ===(V2,NO_PARKING_AVAILABLE_IN,LE3,IS3,180)===> IS3}
15.--->  {Parent to child: IS3 ===(NO_PARKING_AVAILABLE_IN,LE3,IS3,180)===> V2}
16.--->  {Parent to child: ISC ===()===> IS1}
17.--->  {Local: IS1 ===(LE1,6,payable,all,60,10)===> IS1}
18.--->  {Child to parent: IS1 ===(LE1,6,payable,all,60,10)===> ISC}
19.--->  {binding: de -> LE4}
20.--->  {Parent to child: ISC ===()===> IS2}
21.--->  {Local: IS2 ===(LE1,7,payable,all,90,10)===> IS2}
22.--->  {Child to parent: IS2 ===(LE1,7,payable,all,90,10)===> ISC}
23.--->  {Parent to child: ISC ===()===> IS3}
24.--->  {Local: IS3 ===(LE3,8,all,disabled,120,10)===> IS3}
25.--->  {Child to parent: IS3 ===(LE3,8,all,disabled,120,10)===> ISC}
26.--->  {Sibling to sibling: ISC ===(V2,ALTERNATIVE_PARKING_RESERVED_IN,LE3,IS3,120)===> IS3}
```

Table 6.14: The ccaPL output of alternative valid parking Scenario 2

The explanation of the output's syntax in the alternative valid parking scenario 2 is as follows. The ISC starts by calling the database records. The ambient V2 then acts on the capability and moves into the sibling ambient IS2 to become a child of the parent IS2 and V2 starts sending the RM to IS2. The IS2 forwards it to the ISC (sibling to sibling) in order to process the RM and sends the reservation details back to source V2. Lines 6 to 12 show the matching process between the RM and the IS's policies.

This results in the first response which confirms that there is no parking reservation available as requested in the original RM. This leads to the ISC altering the 'duration' parameter. Line 24 shows the new matching process between the RM and the IS's policies. Line 26 confirms the reservation for the alternative parking.

The complete ccaPL code for this scenario can be found in Appendix A.

6.4 Summary

This chapter presented novel, intelligent parking reservation solutions for managing the parking space location and reservation using the concept of VANET and context-aware as a step towards improving the ITS and making the parking reservation experience more comfortable for drivers.

The parking space reservation activity diagram presented in Chapter 5 which shows the work flow between various components of the system from the instant when the driver initiates the parking request through the OBU to when the system allocates and reserves the proper parking space has been eminently validated in this chapter. Here, the ccaPL executable programming language validates the CCA mathematical specification presented in the previous chapter demonstrating the context-aware and mobility capabilities of the system.

Three different experiments have been applied to cover the possible cases. The experiments have shown a real-time simulation and service based on VANET and context aware systems, concluding the system to be efficient and effective; efficient by preventing unnecessary and time-consuming delays, wasting fuel and environmental pollution and traffic congestion; and effective by reducing driver frustration thereby making it a pleasant experience for all road users.

Chapter 7

Conclusion and Future Work

Objectives:

- To summarise the work in this thesis
- To list the main aims achieved
- To propose future work

7.1 Conclusion

The thesis proposes the development and use of an effective on-street parking system which significantly helps to address and alleviate a number of traffic congestion concerns relating to drivers' difficulties in locating an appropriate parking space. There is increasing evidence to suggest that drivers spend a disproportionate time trying to locate a suitable parking space for their vehicle so any system that can reduce this is surely to be welcomed.

It describes utilising the IS based context-aware on-street parking system in order to collect and configure the information to disseminate vacant parking space information to drivers. In doing so, it builds on earlier research done in the design of smart parking systems but focuses on on-street parking spaces as the central topic for research.

Implementing the proposed system will reduce these pressures and enhance the driver experience, particularly in city areas where traffic congestion can be unavoidable and stressful.

The premise of this parking solution as a comfort application is to improve traffic flow thereby preventing time-consuming delays, wasting fuel and pollution, and essentially making the driving a pleasurable experience. These aims were achieved by contributions described in the main body of the thesis.

Chapter 4 introduced the architecture for VANET, based on the concept of a context aware system which facilitates a technique for locating and reserving on-street parking. The vehicle's OBU, IS and ISC are all seen to be working together in order to respond to the request and allocate an appropriate match as per the vehicle and driver profiles by adhering to the IS policy.

Chapter 5 formalised the model in CCA to describe the system ambient and its mobility, context-aware and communication capabilities. The full mathematical specification is represented by the CCA graphical and textual syntax which validates the proposed model making it executable -ready.

In Chapter 6, the ccaPL executable programming language was seen to validate the CCA mathematical specification demonstrating the context-aware capabilities of the system. The three experiments tested the system's efficacy in being able to utilise the available information to cover the main possible scenarios of locating and reserving a vacant parking space.

The inclusion of this application in smart cars alongside the other applications will define the next level in the evolution of the intelligent parking system. This work has created enough scope for modification of the system to provide a more reliable and real-time travel and traffic information in an ethical and safe manner to improve the efficiency of existing parking system that utilises IS and context aware features.

7.2 Achieved Aims

The following aims have been achieved in line with the contributions outlined in Chapter 1:

- An IS-based three tier network topology has been proposed which offers a stable coverage to ensure the packet forwarding among the parking system as a whole.

- The creation of a context-aware based architecture allows for an integrated approach for utilising VANET, IS and context aware systems in a single framework and provides a comprehensive architecture for smart parking system.
- The step-by-step parking space algorithms have been developed to operate the two main functions of locating and reserving an available parking space or searching for an alternative.
- The CCA has been established as a suitable notation for modelling context aware parking system.
- The ccaPL has proved to be an effective validation tool for the purposes of verifying the on-street parking behaviour.

7.3 Future work

A novel and smart idea though this is, there are admittedly a few gaps which could be filled by undertaking a focused approach to future work, some of which will be discussed below.

Firstly, a supplementary piece of future work can be undertaken to explore what would happen in the event where a vehicle were to spot and enter a vacant parking space without making a prior reservation with the IS, and it's related consequences. Any depth of future work addressing this specific scenario would be welcome as the likelihood of such a scenario occurring would be more common than not.

A secondary piece of future work can be carried out which looks at designing mechanisms to locate and reserve nearby parking in case a driver wishes to extend the parking duration. This would occur when an existing vehicle needs to occupy the parking for a longer duration than initially booked, but another vehicle has reserved that parking after the booking duration of the current vehicle expires. The future work would look at designing conflict resolution mechanisms to manage such a potentially incongruous situation in order to meet the needs of both the drivers.

The second piece of future work could be extended to look into the payment system which currently only allows the 'one payment at a time' method of paying for a booking. To extend the parking duration, a driver is required to repeat the same steps which could be unnecessarily inconvenient. The proposed system offers no additional assistance to vehicles already holding a valid parking space so removing this inconvenience could better help serve drivers who may be looking to extend their stay. This idea can be further extended with the development of a Smart Phone application. The smart app can act as the OBU so that the driver can book, reserve or make payment remotely.

Another area of future work can be to explore specific scenarios such as where two cars or more are looking to park in the same parking spot. A protocol will need to be developed to allow the ISC to prioritise one vehicle over the other based on parking zone, vehicle and driver information.

A possible extension of future work could be to devise mechanisms for cross border operations. This would occur where vehicles looking for parking in another country can be traced back to their entry point. This is an important safety application that extends from the work done in this thesis, and can practically serve to identify illegal vehicles in a country.

A final point for future work would be to fully investigate the security implications of the proposed architecture and application. This would ensure that all the VANET-based processes and mechanisms which hold information and deliver services are protected from unauthorised access and illegal activity.

Bibliography

[1] Kopylova Y. Farkas C. *Application Level Protocol for Accident Reconstruction in VANETs*. E-Book, 2007.

[2] D. Jiang, V. Taliwal, A. Meier, W. Holfelder, and R. Herrtwich. Design of 5.9 ghz dsrc-based vehicular safety communication. *Wireless Communications, IEEE*, 13(5):36–43, October 2006.

[3] Yi Yang and Rajive Bagrodia. Evaluation of vanet-based advanced intelligent transportation systems. In *Proceedings of the Sixth ACM International Workshop on VehiculAr InterNETworking*, VANET '09, pages 3–12, New York, NY, USA, 2009. ACM.

[4] M. Tlais and F. Weis. Distributed communication model in hierarchical infostation systems: Research articles. *Concurr. Comput. : Pract. Exper.*, 19(8):1183–1192, June 2007.

[5] H. Chowdhury, J.-P. Makela, and K. Pahlavan. On the random crossing of an infostation coverage. In *Vehicular Technology Conference, 2006. VTC-2006 Fall. 2006 IEEE 64th*, pages 1–5, Sept 2006.

[6] S. Momeni and M. Fathy. Vanet's communication. In *Spread Spectrum Techniques and Applications, 2008. ISSSTA '08. IEEE 10th International Symposium on*, pages 608–612, Aug 2008.

[7] Gruteser M. Oh S., Kang J. Location-based flooding techniques for vehicular emergency messaging sangho oh, 2006.

[8] Ramu Panayappan, Jayini Mukul Trivedi, Ahren Studer, and Adrian Perrig. Vanet-based approach for parking space availability. In *Proceedings of the Fourth ACM International Workshop on Vehicular Ad Hoc Networks*, VANET '07, pages 75–76, New York, NY, USA, 2007. ACM.

[9] Thierry Delot, Nicolas Cenerario, Sergio Ilarri, and Sylvain Lecomte. A cooperative reservation protocol for parking spaces in vehicular ad hoc networks. In *Proceedings of the 6th International Conference on Mobile Technology, Application Systems*, Mobility '09, pages 30:1–30:8, New York, NY, USA, 2009. ACM.

[10] V. Verroios, V. Efstathiou, and A. Delis. Reaching available public parking spaces in urban environments using ad hoc networking. In *Mobile Data Management (MDM), 2011 12th IEEE International Conference on*, volume 1, pages 141–151, June 2011.

[11] Gongjun Yan, Stephan Olariu, Michele C. Weigle, and Mahmoud Abuelela. Smartparking: A secure and intelligent parking system using notice.

[12] Binh An Truong, Young-Koo Lee, and Sung-Young Lee. Modeling and reasoning about uncertainty in context-aware systems. In *e-Business Engineering, 2005. ICEBE 2005. IEEE International Conference on*, pages 102–109, Oct 2005.

[13] M. Al-Sammarraie, F. Siewe, and H. Zedan. Formal specification of an intelligent message notification service in an infostation-based mlearning system using cca. In *Current Trends in Information Technology (CTIT), 2011 International Conference and Workshop on*, pages 64–71, Oct 2011.

[14] R. Souissi, O. Cheikhrouhou, I. Kammoun, and M. Abid. A parking management system using wireless sensor networks. In *Microelectronics (ICM), 2011 International Conference on*, pages 1–7, Dec 2011.

[15] Shihong Qin; Xiangling Yao. An intelligent parking system based on gsm module. *Applied Mathematics and Information Sciences*, 7:55, February 2013.

[16] V.W.S. Tang, Yuan Zheng, and Jiannong Cao. An intelligent car park management system based on wireless sensor networks. In *Pervasive Computing and Applications, 2006 1st International Symposium on*, pages 65–70, Aug 2006.

[17] Jinqi Zhu, Yong Feng, and Bang Liu. Pass: Parking-lot-assisted carpool over vehicular ad hoc networks. *IJDSN*, 2013, 2013.

[18] G. Revathi and V.R.S. Dhulipala. Smart parking systems and sensors: A survey. In *Computing, Communication and Applications (ICCCA), 2012 International Conference on*, pages 1–5, Feb 2012.

[19] Frank Kargl. Sicherheit in mobilen ad hoc netzwerken. Master's thesis, Dissertation Thesis, Ulm, Germany, 10/2003 2003.

[20] Mario Marchese. Wireless pervasive networks for safety operations and secure transportations. In *Wireless Pervasive Computing (ISWPC), 2010 5th IEEE International Symposium on*, pages 226–231, May 2010.

[21] A.H. Al-Bayatti, H. Zedan, and A. Cau. Security solution for mobile ad hoc network of networks (manon). In *Networking and Services, 2009. ICNS '09. Fifth International Conference on*, pages 255–262, April 2009.

[22] DD. Ward, D.A. Topham, C.C. Constantinou, and T.N. Arvanitis. Developments in vehicle-to-vehicle communications. In Jürgen Valldorf and Wolfgang

Gessner, editors, *Advanced Microsystems for Automotive Applications 2005*, Advanced Microsystems for Automotive Applications 2005, pages 353–370. Springer Berlin Heidelberg, 2005.

[23] J. Chinrungrueng, U. Sunantachaikul, and S. Triamlumlerd. Smart parking: An application of optical wireless sensor network. In *Applications and the Internet Workshops, 2007. SAINT Workshops 2007. International Symposium on*, pages 66–66, Jan 2007.

[24] Rongxing Lu, Xiaodong Lin, Haojin Zhu, and Xuemin Shen. Spark: A new vanet-based smart parking scheme for large parking lots. In *INFOCOM 2009, IEEE*, pages 1413–1421, April 2009.

[25] Murat Caliskan, Daniel Graupner, and Martin Mauve. Decentralized discovery of free parking places. In *Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks*, VANET '06, pages 30–39, New York, NY, USA, 2006. ACM.

[26] J.B. Kenney. Dedicated short-range communications (dsrc) standards in the united states. *Proceedings of the IEEE*, 99(7):1162–1182, July 2011.

[27] S. Yousefi, M.S. Mousavi, and M. Fathy. Vehicular ad hoc networks (vanets): Challenges and perspectives. In *ITS Telecommunications Proceedings, 2006 6th International Conference on*, pages 761–766, June 2006.

[28] Alexander Skabardoni Elizabeth Deakin, Karen Trapenberg Frick. Intelligent transport systems: Linking technology and transport policy to help steer the future. 2009.

[29] N. Alam, A.T. Balaie, and A.G. Dempster. Dynamic path loss exponent and distance estimation in a vehicular network using doppler effect and received

signal strength. In *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, pages 1–5, Sept 2010.

[30] H. Moustafa and Y. Zhang. *Vehicular networks: techniques, standards, and applications*. CRC Press, 2009.

[31] Stephan Olariu and Michele C. Weigle. *Vehicular Networks: From Theory to Practice*. Chapman and Hall CRC, 1 edition, 2009.

[32] Trial-use standard for wireless access in vehicular environments (wave) - resource manager. *IEEE Std 1609.1-2006*, pages 1–71, Oct 2006.

[33] Gayathri Chandrasekaran. VANETs: The Networking Platform for Future Vechicular Applications. *Department of Computer Science, Rutgers University*, 2008.

[34] Bin Li, Jian Wang, Tianzhe Dong, and YanHeng Liu. An new approach to access vanets. In *Computing, Communication, Control, and Management, 2009. CCCM 2009. ISECS International Colloquium on*, volume 2, pages 482–485, Aug 2009.

[35] Ieee trial-use standard for wireless access in vehicular environments (wave) - networking services. *IEEE Std 1609.3-2007*, pages 1–99, April 2007.

[36] H. Hartenstein and K. Laberteau. *VANET: Vehicular Applications and Inter-Networking Technologies*. John Wiley and Sons Ltd, 2010.

[37] C. Muller and M. Valle. In-vehicle communication network statistical analysis using vhdl-ams behavioral models. In *Ph.D. Research in Microelectronics and Electronics (PRIME), 2010 Conference on*, pages 1–4, July 2010.

[38] Francois Siewe. ccaPL: a Programming Language for the Calculus of Context-aware Ambients. *Software Technology Research Laboratory, De Montfort University*, March 2012.

[39] Ivan Ganchev, Stanimir Stojanov, Mairtin O'Droma, and Damien Meere. An infostation-based university campus system for the provision of mlearning services. In *Proceedings of the Sixth IEEE International Conference on Advanced Learning Technologies*, ICALT '06, pages 195–199, Washington, DC, USA, 2006. IEEE Computer Society.

[40] Markus Jakobsson, Xiaofeng Wang, and Susanne Wetzel. Stealth attacks on vehicular wireless networks (invited paper).

[41] D. Meere Ganchev I., M. O'Droma. M-learning and m-teaching architectures and the integration of evolving multi-campus educational support e-services. In *M-libraries: libraries on the move to provide virtual access*, pages 159–171. Facet Publishing, UK, 2008.

[42] P. Shah. *Vanet*. LAP Lambert Academic Publishing, 2012.

[43] Mu-Youl Lee and Hyun-Wook Jin. User-level network protocol stacks for automotive infotainment systems. In *Embedded and Ubiquitous Computing (EUC), 2010 IEEE/IFIP 8th International Conference on*, pages 7–14, Dec 2010.

[44] Wai Chen, R.K. Guha, T. Kwon, J. Lee, and I.Y. Hsu. A survey and challenges in routing and data dissemination in vehicular ad-hoc networks. In *Vehicular Electronics and Safety, 2008. ICVES 2008. IEEE International Conference on*, pages 328–333, Sept 2008.

[45] Ekram Hossain, Garland Chow, Victor C. M. Leung, Robert D. McLeod, Jelena Mišić, Vincent W. S. Wong, and Oliver Yang. Vehicular telematics over

heterogeneous wireless networks: A survey. *Comput. Commun.*, 33(7):775–793, May 2010.

[46] H. Fussler, S. Schnaufer, M. Transier, and W. Effelsberg. Vehicular ad-hoc networks: from vision to reality and back. In *Wireless on Demand Network Systems and Services, 2007. WONS '07. Fourth Annual Conference on*, pages 80–83, Jan 2007.

[47] N.F. Abdullah, A Doufexi, and R.J. Piechocki. Spatial diversity for ieee 802.11p post-crash message dissemination in a highway environment. In *Vehicular Technology Conference (VTC 2010-Spring), 2010 IEEE 71st*, pages 1–5, May 2010.

[48] Sayem Chaklader, Junaed Alam, Monirul Islam, and Ali Shihab Sabbir. Black box: An emergency rescue dispatch system for road vehicles for instant notification of road accidents and post crash analysis. In *Informatics, Electronics Vision (ICIEV), 2014 International Conference on*, pages 1–6, May 2014.

[49] A. Schmidt and K. Van Laerhoven. How to build smart appliances? *Personal Communications, IEEE*, 8(4):66–71, Aug 2001.

[50] J. Jakubiak and Y. Koucheryavy. State of the art and research challenges for vanets. In *Consumer Communications and Networking Conference, 2008. CCNC 2008. 5th IEEE*, pages 912–916, Jan 2008.

[51] J. Jakubiak and Y. Koucheryavy. State of the art and research challenges for vanets. In *Consumer Communications and Networking Conference, 2008. CCNC 2008. 5th IEEE*, pages 912–916, Jan 2008.

[52] Nianbo Liu, Ming Liu, Wei Lou, Guihai Chen, and Jiannong Cao. Pva in vanets: Stopped cars are not silent. In *INFOCOM, 2011 Proceedings IEEE*, pages 431–435, April 2011.

[53] Michael J. Marcus. Spectrum issues in fcc's national broadband plan [spectrum policy and regulatory issues]. *Wireless Communications, IEEE*, 17(2):6–6, April 2010.

[54] Trial-use standard for wireless access in vehicular environments (wave) - multi-channel operation. *IEEE Std 1609.4-2006*, pages 1–82, Nov 2006.

[55] Jeroen Hoebeke, Ingrid Moerman, Bart Dhoedt, and Piet Demeester. An overview of mobile ad hoc networks: Applications and challenges. *Journal of the Communications Network*, 3:60–66, July 2004.

[56] M. Caliskan, A. Barthels, B. Scheuermann, and M. Mauve. Predicting parking lot occupancy in vehicular ad hoc networks. In *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, pages 277–281, April 2007.

[57] M. Al-Doori. *Directional Routing Techniques in VANET*. PhD thesis, 2011.

[58] S. Loke. *Context-Aware Pervasive Systems*. Auerbach Publications, 2006.

[59] M. Raya, P. Papadimitratos, and J-P Hubaux. Securing vehicular communications. *Wireless Communications, IEEE*, 13(5):8–15, October 2006.

[60] R.H. Frenkiel, B.R. Badrinath, J. Borres, and R.D. Yates. The infostations challenge: balancing cost and ubiquity in delivering wireless data. *Personal Communications, IEEE*, 7(2):66–71, Apr 2000.

[61] M. Tlais and F. Weis. Data exchanging in hierarchical infostation systems. In *Parallel Processing Workshops, 2007. ICPPW 2007. International Conference on*, pages 10–10, Sept 2007.

[62] Ivan Ganchev, Damien Meere, Stanimir Stojanov, Mícheál Ó. hAodha, and Máirtín O'Droma. On infostation-based mobile services support for library

information systems. In *Proceedings of the 2008 Eighth IEEE International Conference on Advanced Learning Technologies*, ICALT '08, pages 679–681, Washington, DC, USA, 2008. IEEE Computer Society.

[63] Jin Wang, M. Caggiano, and J. Evans. Throughput analysis and rate adaptation for infostation system. In *Vehicular Technology Conference, 2001. VTC 2001 Fall. IEEE VTS 54th*, volume 3, pages 1726–1730 vol.3, 2001.

[64] S.J. Vaughan-Nichols. The challenge of wi-fi roaming. *Computer*, 36(7):17–19, July 2003.

[65] D. Goodman, J. Borras, Narayan B. Mandayam, and R.D. Yates. Infostations: a new system model for data and messaging services. In *Vehicular Technology Conference, 1997, IEEE 47th*, volume 2, pages 969–973 vol.2, May 1997.

[66] J. Borras and R.D. Yates. Highway infostations. WPMC '99, 1999.

[67] A.-R. Rhiemeier and Y. Bar-Ness. Smart antenna systems for infostation data terminals. In *Personal, Indoor and Mobile Radio Communications, 2000. PIMRC 2000. The 11th IEEE International Symposium on*, volume 1, pages 67–71 vol.1, 2000.

[68] John Krumm, editor. *Ubiquitous Computing Fundamentals*. Chapman and Hall/CRC, Boca Raton, FL, 2010.

[69] S.K. Mostefaoui, Z. Maamar, and G.M. Giaglis. *Advances in Ubiquitous Computing: Future Paradigms and Directions*. IGI Global research collection. IGI Pub., 2008.

[70] Uwe Hansmann. *Pervasive computing : the mobile world*. Springer, Berlin New York, 2003.

[71] M. Ilka, M. Niamanesh, and A. Faraahi. A context-aware service discovery in pervasive computing environments. In *Embedded Computing (MECO), 2012 Mediterranean Conference on*, pages 26–29, June 2012.

[72] Gregory D. Abowd, Anind K. Dey, Peter J. Brown, Nigel Davies, Mark Smith, and Pete Steggles. Towards a better understanding of context and context-awareness. In *Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing*, HUC '99, pages 304–307, London, UK, UK, 1999. Springer-Verlag.

[73] B. Xu, A. Ouksel, and O. Wolfson. Opportunistic resource exchange in inter-vehicle ad-hoc networks. In *Mobile Data Management, 2004. Proceedings. 2004 IEEE International Conference on*, pages 4–12, 2004.

[74] Jakob E. Bardram. Hospitals of the future – ubiquitous computing support for medical work in hospitals. In *UbiHealth 2003 the 2nd International Workshop on Ubiquitous Computing for Pervasive Healthcare Applications*, 2003.

[75] Mohammad Obaidat. *Pervasive computing and networking*. John Wiley and Sons, Hoboken, N.J, 2011.

[76] Tom Rodden, Keith Chervest, Nigel Davies, and Alan Dix. Exploiting context in hci design for mobile systems. In *in Workshop on Human Computer Interaction with Mobile Devices*, 1998.

[77] Guanling Chen and David Kotz. A survey of context-aware mobile computing research. Technical report, Hanover, NH, USA, 2000.

[78] Jie Sun, Yongping Zhang, and Jianbo Fan. Towards a context-aware middleware in smart car space. In *Genetic and Evolutionary Computing (ICGEC), 2010 Fourth International Conference on*, pages 276–279, Dec 2010.

[79] Albrecht Schmidt, Michael Beigl, and Hans w. Gellersen. There is more to context than location. *Computers and Graphics*, 23:893–901, 1998.

[80] Valery Naumov, Rainer Baumann, and Thomas Gross. An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces. In *Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '06, pages 108–119, New York, NY, USA, 2006. ACM.

[81] Martin Azizyan, Ionut Constandache, and Romit Roy Choudhury. Surroundsense: Mobile phone localization via ambience fingerprinting. In *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking*, MobiCom '09, pages 261–272, New York, NY, USA, 2009. ACM.

[82] Zhanlin Ji, I. Ganchev, and M. O'Droma. Intelligent software architecture for the service layer of wireless billboard channels. In *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE*, pages 1–2, Jan 2009.

[83] Planning for intelligent transport systems. proposals for the introduction of intelligent transport systems into the 5.9 ghz band in australia. *Australian Communications and Media Authority*, 2009.

[84] Vasant N. Bhonge Manjusha Patil. Wireless Sensor Network and RFID for Smart Parking System. *International Journal of Emerging Technology and Advanced Engineering*, 3:188–192, april 2013.

[85] Magda El Zarki, Sharad Mehrotra, Gene Tsudik, and Nalini Venkatasubramanian. Security issues in a future vehicular network. In *In European Wireless*, pages 270–274, 2002.

[86] Luca Cardelli and Andrew D. Gordon. Mobile ambients. *Theoretical Computer Science*, 240(1):177 – 213, 2000.

[87] M. Al-Sammarraie. *Policy-based Approach for Context-aware Systems*. PhD thesis, 2009.

[88] François Siewe, Hussein Zedan, and Antonio Cau. The calculus of context-aware ambients. *Journal of Computer and System Sciences*, 77(4):597 – 620, 2011. {JCSS} {IEEE} {AINA} 2009.

[89] R. Milner. *Communicating and Mobile Systems: The Pi-Calculus*. Cambridge University Pres, 1999.

[90] D. Sangiorgi and D. Walker. *The PI-Calculus: A Theory of Mobile Processes*. Cambridge University Pres, 2001.

[91] F. Siewe, A. Cau, and H. Zedan. Cca: A calculus of context-aware ambients. In *Advanced Information Networking and Applications Workshops, 2009. WAINA '09. International Conference on*, pages 972–977, May 2009.

[92] Matthias Baldauf, Schahram Dustdar, and Florian Rosenberg. A survey on context aware systems. *Int. J. Ad Hoc Ubiquitous Comput.*, 2(4):263–277, June 2007.

[93] K. Jiang. A sensor guided parallel parking system for nonholonomic vehicles. In *Intelligent Transportation Systems, 2000. Proceedings. 2000 IEEE*, pages 270–275, 2000.

[94] A. El-Rabbany. *Introduction to GPS: The Global Positioning System*. Artech House, 2002.

[95] A. Bastiaansen. The navigable digital street map is the critical success factor for vehicle navigation and transport information systems in europe. In *Intelligent Vehicles Symposium, 1996., Proceedings of the 1996 IEEE*, pages 117–119, Sep 1996.

[96] A. Bastiaansen. Interchangeability standard for digital maps open the market for car navigation systems. In *Intelligent Transportation System, 1997. ITSC '97., IEEE Conference on*, pages 520–524, Nov 1997.

[97] Yong-Qiang Han, Jia-Bin Chen, Zhi-De Liu, Chun lei Song, and Dun-Hui Zhao. A new correction method for dead reckoning navigation system based on digital map. In *Machine Learning and Cybernetics (ICMLC), 2010 International Conference on*, volume 3, pages 1378–1383, July 2010.

[98] Gang Wu, Churng-Wen Chu, K. Wine, J. Evans, and R. Frenkiel. Winmac: a novel transmission protocol for infostations. In *Vehicular Technology Conference, 1999 IEEE 49th*, volume 2, pages 1340–1344 vol.2, Jul 1999.

[99] M. Jerbi, P. Marlier, and S.M. Senouci. Experimental assessment of v2v and i2v communications. In *Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE Internatonal Conference on*, pages 1–6, Oct 2007.

[100] Michele Weigle. Standards: Wave / dsrc / 802.11p. Spring 2008.

[101] M. M. Vianna, L. Portugal, and R. Balassiano. Intelligent transportation systems and parking management: implementation potential in a Brazilian city. *The International Journal of Urban Policy and Planningl*, 21(2):137–148, 2004.

Appendix A

Formal Executable Specification of the On-Street Parking Reservation System

Appendix A contains the full formal executable specification of the proposed on-street parking reservation system. The proposed context-aware based system and the mobile ad-hoc capability are both compatible with the CCA mathematical specifications as asserted in Chapter 5. The programming language for the CCA, known as ccaPL is an executable code of the converted CCA syntax. The ccaPL interpreter is based on the Java platform which can run on any operating system that can support the Java Virtual Machine(JVM).

A.1 The Experiments Scenarios

System: Reserving a vacant parking space

A.1.1 Experiment 1: Valid parking

A.1.1.1 Scenario 1.1

- version: 1.4
- Component:
  - one InfoStation Centre (ISC)
  - many InfoStations (IS1, IS2, ...)
  - many vehicles (V1, V2, ...)
- Behaviour:
  - ISC is fixed, i.e. immobile; and wirely connected to each IS.
  - IS is a wireless access point
  - vehicles can move between ISs. When a vehicle is inside an IS, we say that that vehicle is in range with that IS, and so can communicate wirelessly with that IS.
- Behaviour of ISC:
  - receive request from an IS
  - check request
  - if space available send `Parking_Reserved` to IS
  - else send `No_Parking_Available` to IS

- Format of a request message:
  - is: InfoStation where the sender is located
  - sender: the vehicle sending the request
  - DE: destination, e.g. LE1
  - VS: vehicle size
  - PC: parking cost, e.g. free, payable
  - DT: driver type, e.g. disabled, non-disabled
  - DU: parking duration, e.g. 60 minutes
  - ET: expected time to destination, e.g. 5 minutes
- Master Database:
  - each IS has an entry in the database represented as a memory cell named after that IS.
  - that memory cells stores the following information:

    + ISL: IS location

    + AS: available space

    + ISPC: IS parking cost

    + PS: parking scheme

    + ISDU: parking duration in minutes

    + AT: arrival time in minutes (Driver should arrive within this time)

```
BEGIN_DECLS
  def has(x) = this|x[true]|true
  //display code

END_DECLS

proc memory(IS, ISL, AS, ISPC, PS, ISDU, AT)
```

```
    IS[
send(ISL, AS, ISPC, PS, ISDU, AT).0
      |
        !@recv().recv(ISL1, AS1, ISPC1, PS1, ISDU1, AT1).send(ISL1, AS1, ISPC1, PS1,
ISDU1, AT1).0 | @send(ISL1, AS1, ISPC1, PS1, ISDU1, AT1).0
      | !@recv(ISL2, AS2, ISPC2, PS2, ISDU2, AT2).recv(a, b, c, d, e, f).send(ISL2,
AS2, ISPC2, PS2, ISDU2, AT2).0 |  @send().0
    ]

|

//InfoStation Centre
ISC[
   !::recv(is, sender, DE, VS, PC, DT, DU, ET).let vs = VS+2 in
         IS1#send().IS1#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL and vs <= AS and
(ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU
and AT >= ET>is::send(sender, Parking_Reserved_in, DE, IS1).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and ISDU >= DU and AT >= ET)>IS2#send().IS2#recv(ISL, AS, ISPC, PS, ISDU, AT).
<DE=ISL and vs <= AS and  (ISPC=PC or ISPC=all) and (PS=all or PS=DT) and
ISDU >= DU and AT >= ET>is::send(sender, Parking_Reserved_in, DE, IS2).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or
PS=DT) and ISDU >= DU and
AT >= ET)>IS3#send().IS3#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL and vs <= AS and
(ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and
AT >= ET>is::send(sender, Parking_Reserved_in, DE, IS3).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and ISDU >= DU and AT >= ET)>is::send(sender, No_Parking_Available_in, DE, is).0




  |

//  Master Database start here (InfoStation policies)
    @memory(IS1, LE1, 6, payable, all, 60, 10).0
  | @memory(IS2, LE1, 7, payable, all, 90, 10).0
  | @memory(IS3, LE3, 8, all, disabled, 120, 10).0
// End Master Database
]
```

```
// InfoStations
|
IS1[
   !#recv(sender, DE, VS, PC, DT, DU, ET).ISC::send(IS1, sender, DE, VS, PC,
DT, DU, ET).0
  | ! ISC::recv(sender, reply, dest, info).sender#send(reply, dest, info).0
]
|
IS2[
   !#recv(sender, DE, VS, PC, DT, DU, ET).ISC::send(IS2, sender, DE, VS, PC,
DT, DU, ET).0
  | ! ISC::recv(sender, reply, dest, info).sender#send(reply, dest).0
]
|
IS3[
   !#recv(sender, DE, VS, PC, DT, DU, ET).ISC::send(IS3, sender, DE, VS, PC,
DT, DU, ET).0
  | ! ISC::recv(sender, reply, dest, info).sender#send(reply, dest, info).0
]

// Vehicles
|
V1[
   in IS1.0
  | @send(V1, LE1, 4, payable, non_disabled, 60, 1).@recv(reply, dest, info).0
]
```

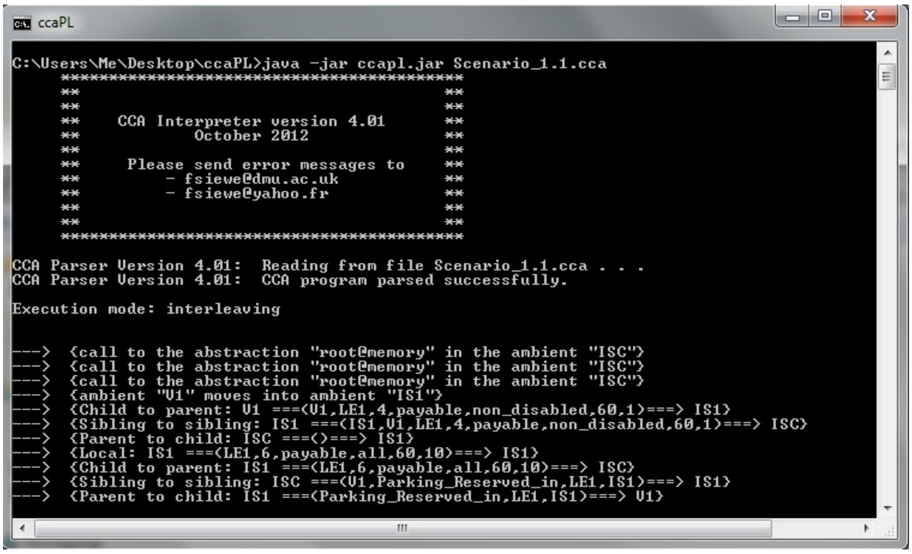


Figure A.1: ccaPL interpreter output of the valid parking Scenario 1

A.1.1.2 Scenario 1.2

- version: 1.4
- Component:
  - one InfoStation Centre (ISC)
  - many InfoStations (IS1, IS2, ...)
  - many vehicles (V1, V2, ...)
- Behaviour:
  - ISC is fixed, i.e. immobile; and wirely connected to each IS.
  - IS is a wireless access point
  - vehicles can move between ISs. When a vehicle is inside an IS, we say that that vehicle is in range with that IS, and so can communicate wirelessly with that IS.
- Behaviour of ISC:

  - receive request from an IS
  - check request
  - if space available send `Parking_Reserved` to IS
  - else send `No_Parking_Available` to IS

- Format of a request message:
  - is: InfoStation where the sender is located
  - sender: the vehicle sending the request
  - DE: destination, e.g. LE1
  - VS: vehicle size
  - PC: parking cost, e.g. free, payable
  - DT: driver type, e.g. disabled, non-disabled
  - DU: parking duration, e.g. 60 minutes
  - ET: expected time to destination, e.g. 5 minutes

- Master Database:
  - each IS has an entry in the database represented as a memory cell named after that IS.
  - that memory cells stores the following information:

    + ISL: IS location

    + AS: available space

    + ISPC: IS parking cost

    + PS: parking scheme

    + ISDU: parking duration in minutes

    + AT: arrival time in minutes (Driver should arrive within this time)

```
BEGIN_DECLS
  def has(x) = this|x[true]|true
  //display code

END_DECLS

proc memory(IS, ISL, AS, ISPC, PS, ISDU, AT)
    IS[
send(ISL, AS, ISPC, PS, ISDU, AT).0
      |
        !@recv().recv(ISL1, AS1, ISPC1, PS1, ISDU1, AT1).send(ISL1, AS1, ISPC1,
PS1, ISDU1, AT1).0 | @send(ISL1, AS1, ISPC1, PS1, ISDU1, AT1).0
      | !@recv(ISL2, AS2, ISPC2, PS2, ISDU2, AT2).recv(a, b, c, d, e, f).send(ISL2, AS2,
ISPC2, PS2, ISDU2, AT2).0 | @send().0
    ]

|

//InfoStation Centre
ISC[
   !::recv(is, sender, DE, VS, PC, DT, DU, ET).let vs = VS+2 in
         IS1#send().IS1#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL and vs <= AS and
(ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and
AT >= ET>is::send(sender, Parking_Reserved_in, DE, IS1).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all
or PS=DT) and ISDU >= DU and AT >= ET)>IS2#send().IS2#recv(ISL, AS, ISPC, PS, ISDU, AT)
.<DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and
AT >= ET>is::send(sender, Parking_Reserved_in, DE, IS2).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and ISDU >= DU and AT >= ET)>IS3#send().IS3#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL
and vs <= AS and (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU
and AT >= ET>is::send(sender, Parking_Reserved_in, DE, IS3).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and ISDU >= DU and AT >= ET)>is::send(sender, No_Parking_Available_in,DE, is).0




  |

//  Master Database start here (InfoStation policies)
```

```
    @memory(IS1, LE1, 6, payable, all, 60, 10).0
  | @memory(IS2, LE1, 7, payable, all, 90, 10).0
  | @memory(IS3, LE3, 8, all, disabled, 120, 10).0
// End Master Database
]

// InfoStations
|
IS1[
   !#recv(sender, DE, VS, PC, DT, DU, ET).ISC::send(IS1, sender, DE, VS, PC, DT, DU, ET).0
  | ! ISC::recv(sender, reply, dest, info).sender#send(reply, dest, info).0
]
|
IS2[
   !#recv(sender, DE, VS, PC, DT, DU, ET).ISC::send(IS2, sender, DE, VS, PC, DT, DU, ET).0
  | ! ISC::recv(sender, reply, dest, info).sender#send(reply, dest).0
]
|
IS3[
   !#recv(sender, DE, VS, PC, DT, DU, ET).ISC::send(IS3, sender, DE, VS, PC, DT, DU, ET).0
  | ! ISC::recv(sender, reply, dest, info).sender#send(reply, dest, info).0
]

// Vehicles
|
V2[
   in IS2.0
  | @send(V2, LE3, 6, free, disabled, 60, 10).@recv(reply, dest, info).0
]
```

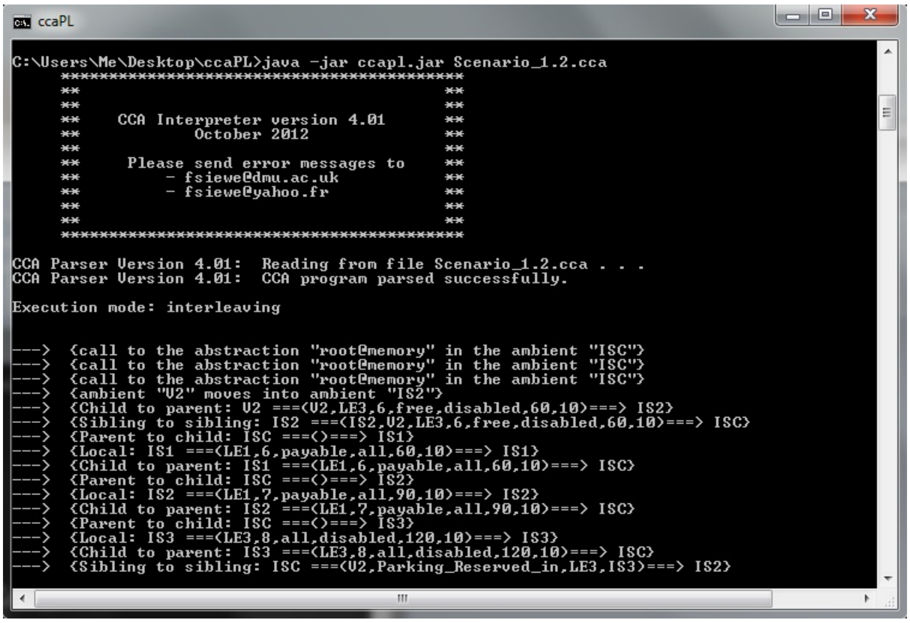


Figure A.2: ccaPL interpreter output of the valid parking Scenario 2

A.1.2 Experiment 2: Non-valid parking

A.1.2.1 Scenario 2.1

- version: 1.4
- Component:
  - one InfoStation Centre (ISC)
  - many InfoStations (IS1, IS2, ...)
  - many vehicles (V1, V2, ...)
- Behaviour:
  - ISC is fixed, i.e. immobile; and wirely connected to each IS.
  - IS is a wireless access point
  - vehicles can move between ISs. When a vehicle is inside an IS, we say that that vehicle is in range with that IS, and so can communicate wirelessly

with that IS.

- Behaviour of ISC:
  - receive request from an IS
  - check request
  - if space available send `Parking_Reserved` to IS
  - else send `No_Parking_Available` to IS
- Format of a request message:
  - is: InfoStation where the sender is located
  - sender: the vehicle sending the request
  - DE: destination, e.g. LE1
  - VS: vehicle size
  - PC: parking cost, e.g. free, payable
  - DT: driver type, e.g. disabled, non-disabled
  - DU: parking duration, e.g. 60 minutes
  - ET: expected time to destination, e.g. 5 minutes
- Master Database:
  - each IS has an entry in the database represented as a memory cell named after that IS.
  - that memory cells stores the following information:

    + ISL: IS location

    + AS: available space

    + ISPC: IS parking cost

+ PS: parking scheme

+ ISDU: parking duration in minutes

+ AT: arrival time in minutes (Driver should arrive within this time)

```
BEGIN_DECLS
  def has(x) = this|x[true]|true
  //display code

END_DECLS

proc memory(IS, ISL, AS, ISPC, PS, ISDU, AT)
    IS[
send(ISL, AS, ISPC, PS, ISDU, AT).0
      |
        !@recv().recv(ISL1, AS1, ISPC1, PS1, ISDU1, AT1).send(ISL1, AS1, ISPC1, PS1,
ISDU1, AT1).0 | @send(ISL1, AS1, ISPC1, PS1, ISDU1, AT1).0
      | !@recv(ISL2, AS2, ISPC2, PS2, ISDU2, AT2).recv(a, b, c, d, e, f).send(ISL2, AS2,
ISPC2, PS2, ISDU2, AT2).0 | @send().0
    ]

|

//InfoStation Centre
ISC[
   !::recv(is, sender, DE, VS, PC, DT, DU, ET).let vs = VS+2 in
         IS1#send().IS1#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL and vs <= AS and
(ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and
AT >= ET>is::send(sender, Parking_Reserved_in, DE, IS1).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and ISDU >= DU and AT >= ET)>IS2#send().IS2#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL
and vs <= AS and (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and
AT >= ET>is::send(sender, Parking_Reserved_in, DE, IS2).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and ISDU >= DU and AT >= ET)>IS3#send().IS3#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL
and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and
AT >= ET>is::send(sender, Parking_Reserved_in, DE, IS3).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and ISDU >= DU and AT >= ET)>is::send(sender, No_Parking_Available_in, DE, Area).0
```

```
  |

//  Master Database start here (InfoStation policies)
    @memory(IS1, LE1, 6, payable, all, 60, 10).0
  | @memory(IS2, LE1, 7, payable, all, 90, 10).0
  | @memory(IS3, LE3, 8, all, disabled, 120, 10).0
// End Master Database
]

// InfoStations
|
IS1[
   !#recv(sender, DE, VS, PC, DT, DU, ET).ISC::send(IS1, sender, DE, VS, PC, DT, DU, ET).0
  | ! ISC::recv(sender, reply, dest, info).sender#send(reply, dest, info).0
]
|
IS2[
   !#recv(sender, DE, VS, PC, DT, DU, ET).ISC::send(IS2, sender, DE, VS, PC, DT, DU, ET).0
  | ! ISC::recv(sender, reply, dest, info).sender#send(reply, dest).0
]
|
IS3[
   !#recv(sender, DE, VS, PC, DT, DU, ET).ISC::send(IS3, sender, DE, VS, PC, DT, DU, ET).0
  | ! ISC::recv(sender, reply, dest, info).sender#send(reply, dest, info).0
]

// Vehicles
|
V1[
   in IS1.0
  | @send(V1, LE2, 4, free, non_disabled, 60, 10).@recv(reply, dest, info).0
]
```

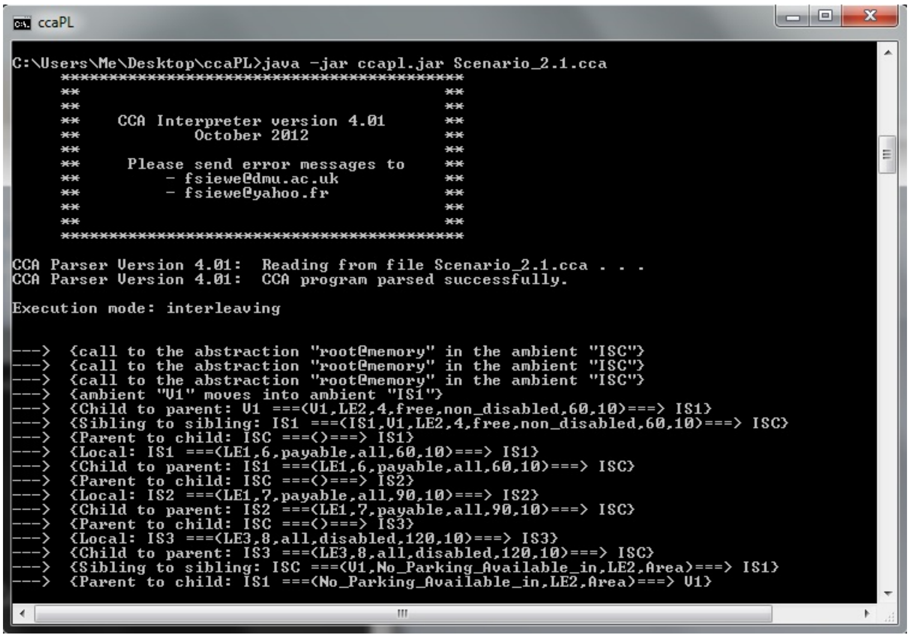


Figure A.3: ccaPL interpreter output of the non-valid parking Scenario 1

A.1.2.2 Scenario 2.2

- version: 1.4
- Component:
  - one InfoStation Centre (ISC)
  - many InfoStations (IS1, IS2, ...)
  - many vehicles (V1, V2, ...)
- Behaviour:
  - ISC is fixed, i.e. immobile; and wirely connected to each IS.
  - IS is a wireless access point
  - vehicles can move between ISs. When a vehicle is inside an IS, we say that that vehicle is in range with that IS, and so can communicate wirelessly with that IS.

- Behaviour of ISC:
  - receive request from an IS
  - check request
  - if space available send `Parking_Reserved` to IS
  - else send `No_Parking_Available` to IS
- Format of a request message:
  - is: InfoStation where the sender is located
  - sender: the vehicle sending the request
  - DE: destination, e.g. LE1
  - VS: vehicle size
  - PC: parking cost, e.g. free, payable
  - DT: driver type, e.g. disabled, non-disabled
  - DU: parking duration, e.g. 60 minutes
  - ET: expected time to destination, e.g. 5 minutes
- Master Database:
  - each IS has an entry in the database represented as a memory cell named after that IS.
  - that memory cells stores the following information:

    + ISL: IS location

    + AS: available space

    + ISPC: IS parking cost

    + PS: parking scheme

+ ISDU: parking duration in minutes

+ AT: arrival time in minutes (Driver should arrive within this time)

```
BEGIN_DECLS
  def has(x) = this|x[true]|true
  //display code

END_DECLS


proc memory(IS, ISL, AS, ISPC, PS, ISDU, AT)
    IS[
send(ISL, AS, ISPC, PS, ISDU, AT).0
      |
        !@recv().recv(ISL1, AS1, ISPC1, PS1, ISDU1, AT1).send(ISL1, AS1, ISPC1, PS1,
ISDU1, AT1).0 | @send(ISL1, AS1, ISPC1, PS1, ISDU1, AT1).0
      | !@recv(ISL2, AS2, ISPC2, PS2, ISDU2, AT2).recv(a, b, c, d, e, f).send(ISL2, AS2,
ISPC2, PS2, ISDU2, AT2).0 | @send().0
    ]


|


//InfoStation Centre
ISC[
   !::recv(is, sender, DE, VS, PC, DT, DU, ET).let vs = VS+2 in
         IS1#send().IS1#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL and vs <= AS and
(ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and
AT >= ET>is::send(sender, Parking_Reserved_in, DE, IS1).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and
ISDU >= DU and AT >= ET)>IS2#send().IS2#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL and
vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and
AT >= ET>is::send(sender, Parking_Reserved_in, DE, IS2).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and ISDU >= DU and AT >= ET)>IS3#send().IS3#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL and vs <= AS
and  (ISPC= PC or ISPC=all) and (PS=all or
PS=DT) and ISDU >= DU and AT >= ET>is::send(sender, Parking_Reserved_in, DE, IS3).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and ISDU >= DU and AT >= ET)>is::send(sender, No_Parking_Available_in, DE, is).0
```

```
  |

//  Master Database start here (InfoStation policies)
    @memory(IS1, LE1, 6, payable, all, 60, 10).0
  | @memory(IS2, LE1, 7, payable, all, 90, 10).0
  | @memory(IS3, LE3, 8, all, disabled, 120, 10).0
// End Master Database
]

// InfoStations
|
IS1[
   !#recv(sender, DE, VS, PC, DT, DU, ET).ISC::send(IS1, sender, DE, VS, PC, DT, DU, ET).0
  | ! ISC::recv(sender, reply, dest, info).sender#send(reply, dest, info).0
]
|
IS2[
   !#recv(sender, DE, VS, PC, DT, DU, ET).ISC::send(IS2, sender, DE, VS, PC, DT, DU, ET).0
  | ! ISC::recv(sender, reply, dest, info).sender#send(reply, dest, info).0
]
|
IS3[
   !#recv(sender, DE, VS, PC, DT, DU, ET).ISC::send(IS3, sender, DE, VS, PC, DT, DU, ET).0
  | ! ISC::recv(sender, reply, dest, info).sender#send(reply, dest, info).0
]

// Vehicles
|
V2[
   in IS3.0
  | @send(V2, LE3, 5, free, non_disabled, 60, 1).@recv(reply, dest, info).0
]
```

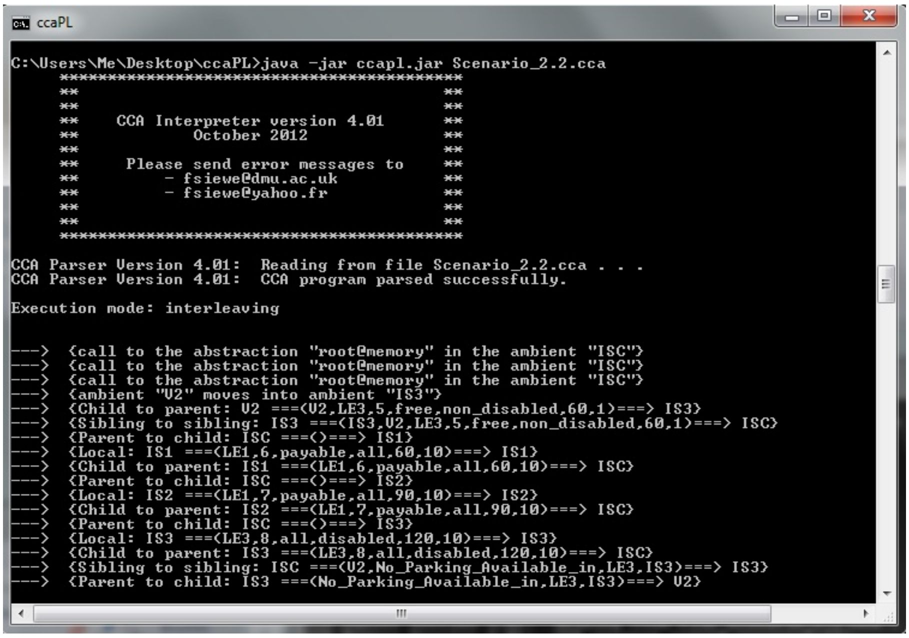


Figure A.4: ccaPL interpreter output of the non-valid parking Scenario 2

A.1.3 Experiment 3: Alternative valid parking

A.1.3.1 Scenario 3.1

- version: 1.4
- Component:
  - one InfoStation Centre (ISC)
  - many InfoStations (IS1, IS2, ...)
  - many vehicles (V1, V2, ...)
- Behaviour:
  - ISC is fixed, i.e. immobile; and wirely connected to each IS.
  - IS is a wireless access point
  - vehicles can move between ISs. When a vehicle is inside an IS, we say that that vehicle is in range with that IS, and so can communicate wirelessly

with that IS.

- Behaviour of ISC:
  - receive request from an IS
  - check request
  - If driver preferences non-flexible

    + if space available send `Parking_Reserved` to IS

    + else send `No_Parking_Available` to IS
  - else

    + if alternative space available send `Alternative_Parking_Reserved` to IS

    + else send `No_Alternative_Parking` to IS
- Format of a request message:
  - is: InfoStation where the sender is located
  - sender: the vehicle sending the request
  - DE: destination, e.g. LE1
  - VS: vehicle size
  - PC: parking cost, e.g. free, payable
  - DT: driver type, e.g. disabled, non-disabled
  - DU: parking duration, e.g. 60 minutes
  - ET: expected time to destination, e.g. 5 minutes
  - FL: driver preferences flexibility, e.g. ISC can change the DE or DU to find an alternative options
- Master Database:

- each IS has an entry in the database represented as a memory cell named after that IS.
- that memory cells stores the following information: + ISL: IS location

  + AS: available space

  + ISPC: IS parking cost

  + PS: parking scheme

  + ISDU: parking duration in minutes

  + AT: arrival time in minutes (Driver should arrive within this time)

- Neighbourhood Database
  - each IS has neighbourhood ISs (parking zones)
  - these neighbourhood can be used as an alternative parking zones when the search diameter expand.

```
BEGIN_DECLS
  def has(x) = this|x[true]|true
  def at2(n,m) = n[m[true] | true] | true // ambient m is a child ambient of n.
  //display code

END_DECLS

proc memory(IS, ISL, AS, ISPC, PS, ISDU, AT)
    IS[
send(ISL, AS, ISPC, PS, ISDU, AT).0
      |
        !@recv().recv(ISL1, AS1, ISPC1, PS1, ISDU1, AT1).send(ISL1, AS1, ISPC1,
PS1, ISDU1, AT1).0 | @send(ISL1, AS1, ISPC1, PS1, ISDU1, AT1).0
      | !@recv(ISL2, AS2, ISPC2, PS2, ISDU2, AT2).recv(a, b, c, d, e, f).send(ISL2,
AS2, ISPC2, PS2, ISDU2, AT2).0 | @send().0
    ]

|

//InfoStation Centre
```

```
ISC[
   !::recv(is, sender, DE, VS, PC, DT, DU, ET, FL). let vs=VS+2  in
        IS1#send().IS1#recv(ISL, AS, ISPC, PS, ISDU, AT).

<DE=ISL and vs <= AS and (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and
AT >= ET>is::send(sender, PARKING_RESERVED_IN, DE, IS1, DU).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and
ISDU >= DU and AT >= ET)>IS2#send().IS2#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL and vs <= AS and
(ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU
and AT >= ET>is::send(sender, PARKING_RESERVED_IN, DE, IS2, DU).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and
ISDU >= DU and AT >= ET)>IS3#send().IS3#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL and vs <= AS
and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and
AT >= ET>is::send(sender, PARKING_RESERVED_IN, DE, IS3, DU).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and
ISDU >= DU and AT >= ET)>is::send(sender, NO_PARKING_AVAILABLE_IN,DE, is, DU).
IS1#send().IS1#recv(ISL, AS, ISPC, PS, ISDU, AT).find de:(somewhere (at2(DE,de))
and FL=yes) for let du=DU/2 in

<(DE=ISL or de=ISL) and vs <= AS and (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and
(ISDU>= DU or ISDU >= du) and DU < ISDU and
AT >= ET>is::send(sender, ALTERNATIVE_PARKING_RESERVED_IN, ISL, IS1, DU).0
| <(DE=ISL or de=ISL) and vs <= AS and (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and
(ISDU>= DU or ISDU >= du)  and DU >= ISDU and
AT >= ET>is::send(sender, ALTERNATIVE_PARKING_RESERVED_IN, ISL, IS1, ISDU).0
| <not ((DE=ISL or de=ISL)  and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and (ISDU>= DU or ISDU >= du) and AT >= ET)>IS2#send().IS2#recv(ISL, AS, ISPC, PS, ISDU, AT)
.<(DE=ISL or de=ISL)  and vs <= AS and (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and
(ISDU>= DU or ISDU >= du) and DU < ISDU and AT >= ET>is::send(sender,
ALTERNATIVE_PARKING_RESERVED_IN, ISL, IS2, DU).0
| <(DE=ISL or de=ISL)  and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and (ISDU>= DU or ISDU >= du) and DU >= ISDU and AT >= ET>is::send(sender,
ALTERNATIVE_PARKING_RESERVED_IN, ISL, IS2, ISDU).0
| <not ((DE=ISL or de=ISL)  and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and (ISDU>= DU or ISDU >= du) and AT >= ET)>IS3#send().IS3#recv(ISL, AS, ISPC, PS, ISDU, AT)
.<(DE=ISL or de=ISL)  and vs <= AS and (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and
(ISDU>= DU or ISDU >= du) and DU < ISDU and AT >= ET>is::send(sender,
ALTERNATIVE_PARKING_RESERVED_IN, ISL, IS3, DU).0
| <(DE=ISL or de=ISL)  and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and (ISDU>= DU or ISDU >= du) and DU >= ISDU and AT >= ET>is::send(sender,
ALTERNATIVE_PARKING_RESERVED_IN, ISL, IS3, ISDU).0
```

```
| <not ((DE=ISL or de=ISL)  and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or
PS=DT) and (ISDU>= DU or ISDU >= du) and AT >= ET)>is::send(sender,
NO_ALTERNATIVE_PARKING_IN,DE, is, DU).0




  |



//  Master Database start here (InfoStation policies)
    @memory(IS1, LE1, 6, payable, all, 60, 10).0
  | @memory(IS2, LE1, 7, payable, all, 90, 10).0
  | @memory(IS3, LE3, 8, all, disabled, 120, 10).0
// End Master Database

// Neighbourhood DB
|
LE1[
  LE2[0]
]
|
LE2[
  LE3[0]
]
|
LE3[
  LE4[0]
]
//end Neighbourhood DB
]

// InfoStations
|
IS1[
   !#recv(sender, DE, VS, PC, DT, DU, ET, FL).ISC::send(IS1, sender, DE, VS, PC, DT, DU, ET, FL).0
  | ! ISC::recv(sender, reply, dest, info, dur).sender#send(reply, dest, info, dur).0
```

```
]
|
IS2[
   !#recv(sender, DE, VS, PC, DT, DU, ET, FL).ISC::send(IS2, sender, DE, VS, PC, DT, DU, ET, FL).0
  | ! ISC::recv(sender, reply, dest, info, dur).sender#send(reply, dest, info, dur).0
]
|
IS3[
   !#recv(sender, DE, VS, PC, DT, DU, ET, FL).ISC::send(IS3, sender, DE, VS, PC, DT, DU, ET, FL).0
  | ! ISC::recv(sender, reply, dest, info, dur).sender#send(reply, dest, info, dur).0
]

// Vehicles
|
V1[
   in IS1.0
  | @send(V1, LE2, 4, free, disabled, 60, 10,yes).@recv(reply, dest, info, dur).0
]
```

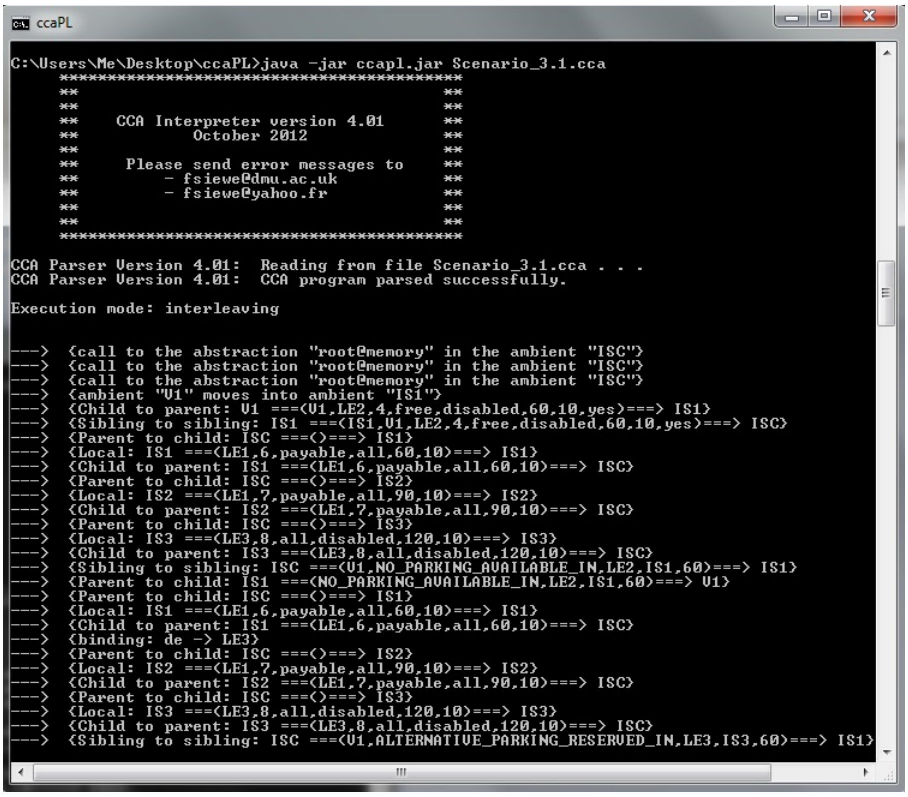


Figure A.5: ccaPL interpreter output of the alternative valid parking Scenario 1

A.1.3.2 Scenario 3.2

- version: 1.4
- Component:
  - one InfoStation Centre (ISC)
  - many InfoStations (IS1, IS2, ...)
  - many vehicles (V1, V2, ...)
- Behaviour:
  - ISC is fixed, i.e. immobile; and wirely connected to each IS.
  - IS is a wireless access point
  - vehicles can move between ISs. When a vehicle is inside an IS, we say that that vehicle is in range with that IS, and so can communicate wirelessly with that IS.
- Behaviour of ISC:
  - receive request from an IS
  - check request
  - If driver preferences non-flexible

    + if space available send `Parking_Reserved` to IS

    + else send `No_Parking_Available` to IS
  - else

    + if alternative space available send `Alternative_Parking_Reserved` to IS

    + else send `No_Alternative_Parking` to IS

- Format of a request message:
  - is: InfoStation where the sender is located
  - sender: the vehicle sending the request
  - DE: destination, e.g. LE1
  - VS: vehicle size
  - PC: parking cost, e.g. free, payable
  - DT: driver type, e.g. disabled, non-disabled
  - DU: parking duration, e.g. 60 minutes
  - ET: expected time to destination, e.g. 5 minutes
  - FL: driver preferences flexibility, e.g. ISC can change the DE or DU to find an alternative options
- Master Database:
  - each IS has an entry in the database represented as a memory cell named after that IS.
  - that memory cells stores the following information: + ISL: IS location

    + AS: available space

    + ISPC: IS parking cost

    + PS: parking scheme

    + ISDU: parking duration in minutes

    + AT: arrival time in minutes (Driver should arrive within this time)
- Neighbourhood Database
  - each IS has neighbourhood ISs (parking zones)

– these neighbourhood can be used as an alternative parking zones when the search diameter expand.

```
BEGIN_DECLS
  def has(x) = this|x[true]|true
  def at2(n,m) = n[m[true] | true] | true // ambient m is a child ambient of n.
  //display code

END_DECLS

proc memory(IS, ISL, AS, ISPC, PS, ISDU, AT)
    IS[
send(ISL, AS, ISPC, PS, ISDU, AT).0
      |
        !@recv().recv(ISL1, AS1, ISPC1, PS1, ISDU1, AT1).send(ISL1, AS1, ISPC1, PS1,
ISDU1, AT1).0 | @send(ISL1, AS1, ISPC1, PS1, ISDU1, AT1).0
      | !@recv(ISL2, AS2, ISPC2, PS2, ISDU2, AT2).recv(a, b, c, d, e, f).send(ISL2, AS2,
ISPC2, PS2, ISDU2, AT2).0 | @send().0
    ]

|

//InfoStation Centre
ISC[
   !::recv(is, sender, DE, VS, PC, DT, DU, ET, FL). let vs=VS+2  in
        IS1#send().IS1#recv(ISL, AS, ISPC, PS, ISDU, AT).

<DE=ISL and vs <= AS and (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and
AT >= ET>is::send(sender, PARKING_RESERVED_IN, DE, IS1, DU).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and ISDU >= DU and AT >= ET)>IS2#send().IS2#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL and
vs <= AS and (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and AT >= ET>is::send
(sender, PARKING_RESERVED_IN, DE, IS2, DU).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and ISDU >= DU and AT >= ET)>IS3#send().IS3#recv(ISL, AS, ISPC, PS, ISDU, AT).<DE=ISL and
vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and ISDU >= DU and AT >= ET>is::send
(sender, PARKING_RESERVED_IN, DE, IS3, DU).0
              | <not (DE=ISL and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and ISDU >= DU and AT >= ET)>is::send(sender, NO_PARKING_AVAILABLE_IN,DE, is, DU).IS1#send()
.IS1#recv(ISL, AS, ISPC, PS, ISDU, AT).find de:(somewhere (at2(DE,de)) and FL=yes)
for let du=DU/2 in
```

```
<(DE=ISL or de=ISL) and vs <= AS and (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and (ISDU>= DU or ISDU >= du) and DU < ISDU and AT >= ET>is::send(sender,
ALTERNATIVE_PARKING_RESERVED_IN, ISL, IS1, DU).0
| <(DE=ISL or de=ISL) and vs <= AS and (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and (ISDU>= DU or ISDU >= du)  and DU >= ISDU and AT >= ET>is::send(sender,
ALTERNATIVE_PARKING_RESERVED_IN, ISL, IS1, ISDU).0
| <not ((DE=ISL or de=ISL)  and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and (ISDU>= DU or ISDU >= du) and AT >= ET)>IS2#send().IS2#recv(ISL, AS, ISPC, PS, ISDU, AT)
.<(DE=ISL or de=ISL)  and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and
(ISDU>= DU or ISDU >= du) and DU < ISDU and AT >= ET>is::send(sender,
ALTERNATIVE_PARKING_RESERVED_IN, ISL, IS2, DU).0
| <(DE=ISL or de=ISL)  and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and
(ISDU>= DU or ISDU >= du) and DU >= ISDU and AT >= ET>is::send(sender,
ALTERNATIVE_PARKING_RESERVED_IN, ISL, IS2, ISDU).0
| <not ((DE=ISL or de=ISL)  and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and (ISDU>= DU or ISDU >= du) and AT >= ET)>IS3#send().IS3#recv(ISL, AS, ISPC, PS, ISDU, AT)
.<(DE=ISL or de=ISL)  and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and (ISDU>= DU or ISDU >= du) and DU < ISDU and AT >= ET>is::send(sender,
ALTERNATIVE_PARKING_RESERVED_IN, ISL, IS3, DU).0
| <(DE=ISL or de=ISL)  and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT) and
(ISDU>= DU or ISDU >= du) and DU >= ISDU and AT >= ET>is::send(sender,
ALTERNATIVE_PARKING_RESERVED_IN, ISL, IS3, ISDU).0
| <not ((DE=ISL or de=ISL)  and vs <= AS and  (ISPC= PC or ISPC=all) and (PS=all or PS=DT)
and (ISDU>= DU or ISDU >= du) and AT >= ET)>is::send(sender, NO_ALTERNATIVE_PARKING_IN,
DE, is, DU).0

  |

//  Master Database start here (InfoStation policies)
    @memory(IS1, LE1, 6, payable, all, 60, 10).0
  | @memory(IS2, LE1, 7, payable, all, 90, 10).0
  | @memory(IS3, LE3, 8, all, disabled, 120, 10).0
```

```
// End Master Database

// Neighbourhood DB
|
LE1[
  LE2[0]
]
|
LE2[
  LE3[0]
]
|
LE3[
  LE4[0]
]
//end Neighbourhood DB
]

// InfoStations
|
IS1[
   !#recv(sender, DE, VS, PC, DT, DU, ET, FL).ISC::send(IS1, sender, DE, VS, PC, DT, DU, ET, FL).0
  | ! ISC::recv(sender, reply, dest, info, dur).sender#send(reply, dest, info, dur).0
]
|
IS2[
   !#recv(sender, DE, VS, PC, DT, DU, ET, FL).ISC::send(IS2, sender, DE, VS, PC, DT, DU, ET, FL).0
  | ! ISC::recv(sender, reply, dest, info, dur).sender#send(reply, dest, info, dur).0
]
|
IS3[
   !#recv(sender, DE, VS, PC, DT, DU, ET, FL).ISC::send(IS3, sender, DE, VS, PC, DT, DU, ET, FL).0
  | ! ISC::recv(sender, reply, dest, info, dur).sender#send(reply, dest, info, dur).0
]

// Vehicles
|
V2[
   in IS3.0
  | @send(V2, LE3, 6, free, disabled, 180, 10,yes).@recv(reply, dest, info, dur).0
]
```

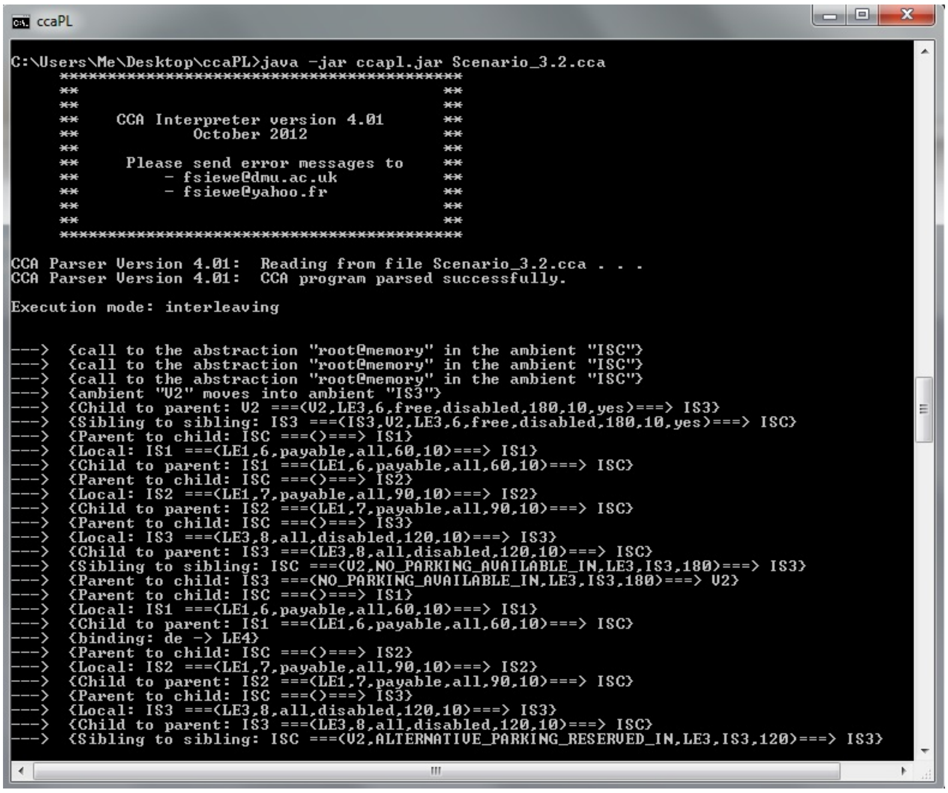


Figure A.6: ccaPL interpreter output of the alternative valid parking Scenario 2