

USER EXPERIENCE-BASED PROVISIONING SERVICES IN VEHICULAR CLOUDS

By

MOAYAD ALOQAILY

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
in Partial Fulfillment of the Requirements
for the Doctorate in Philosophy Degree in Electrical and Computer
Engineering

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

ABSTRACT

Today, the increasing number of applications based on the Internet of Things, as well as advances in wireless communication, information and communication technology, and mobile cloud computing have allowed users to access a wide range of resources while mobile. Vehicular clouds are considered key elements for today's intelligent transportation systems. They are outfitted with equipment to enable applications and services for vehicle drivers, surrounding vehicles, pedestrians and third parties.

As vehicular cloud computing has become more popular, due to its ability to improve driver and vehicle safety and provide provisioning services and applications, researchers and industry have growing interest in the design and development of vehicular networks for emerging applications. Though vehicle drivers can now access a variety of on-demand resources en route via vehicular network service providers, the development of vehicular cloud provisioning services has many challenges. In this dissertation, we examine the most critical provisioning service challenges drivers face, including, cost, privacy and latency. To this point, very little research has addressed these issues from the driver perspective. Privacy and service latency are certainly emerging challenges for drivers, as are service costs since this is a relatively new financial concept.

Motivated by the Quality of Experience paradigm and the concept of the Trusted Third Party, we identify and investigate these challenges and examine the limitations and requirements of a vehicular environment. We found no research that addressed these challenges simultaneously, or investigated their effect on one another. We have developed a Quality of Experience framework that provides scalability and reduces congestion overhead for users. Furthermore, we propose two theory-based frameworks to manage on-demand service provision in vehicular clouds: Auction-driven Multi-objective Provisioning and a Multiagent/Multiobjective Interaction Game System. We present different approaches to these, and show through analytical and simulation results that our potential schemes help drivers minimize costs and latency, and maximize privacy.

DEDICATION

*To the pure spirit of my **father**, who had dedicated his whole life to his family, To my **mother**, for her absolute and endless love. You were my great source of inspiration, prayers, love, support, and my guiding light.*

ACKNOWLEDGMENT

My most sincere appreciation and gratitude go to my advisor, Professor Hussein T. Mouftah, for his kind help, intellectual and personal support, encouragement and patience in the success of my work as a researcher. Professor Mouftah is not only a thesis advisor, but also a role model to me; without his continued support, encouragement, and especially his guidance and mentoring, it would not have been possible to complete this work.

I am sincerely grateful to all the people who have supported me throughout the course of my Ph.D. program especially my mentor Professor Burak Kantarci for his precious time, effort, and constructive feedback.

I would also like to extend my thanks to all members of my examining committee for taking the time and effort to review my thesis.

I have also known many wonderful friends in Ottawa, and although I am not able to mention each of their names, I would like to extend my thanks to all of them.

To my parents, who made all of this possible, for their endless encouragement and support, my success is yours. My high regards are addressed to my parents-in-law as well.

I would like to express my gratitude to my beloved wife Safa for her support and all the sacrifices she has made.

Finally, to my beloved sons, Rawad and Mohammad, you are the joy of my life.

TABLE OF CONTENTS

ABSTRACT	II
DEDICATION.....	III
ACKNOWLEDGMENT	IV
LIST OF FIGURES	IX
LIST OF TABLES	XI
LIST OF ACRONYMS	XII
LIST OF SYMBOLS	XIV
CHAPTER 1 INTRODUCTION.....	1
1.1 OVERVIEW	1
1.2 MOTIVATION.....	2
1.3 OBJECTIVES	4
1.4 THESIS CONTRIBUTIONS	5
1.5 THESIS OUTLINE	5
1.6 LIST OF PUBLICATIONS	6
CHAPTER 2 VEHICULAR CLOUD COMPUTING - STATE OF THE ART.....	8
2.1 INTRODUCTION	8
2.2 VEHICULAR CLOUD COMPUTING (VCC).....	10
2.2.1 TAXONOMY OF VEHICULAR CLOUD COMPUTING.....	11
2.2.2 VEHICULAR CLOUD ARCHITECTURE	12
2.2.3 ADVANTAGES OF VEHICULAR CLOUD COMPUTING	15
2.3 OPEN ISSUES AND CHALLENGES IN VEHICULAR CLOUDS AND EXISTING SOLUTIONS	15
2.3.1 SECURITY AND PRIVACY	16
2.3.2 INFORMATION REDUNDANCY.....	20

2.3.3 IDLE ON-BOARD EQUIPMENT.....	21
2.3.4 INTEROPERABILITY	21
2.3.5 MOBILITY	22
2.3.6 SERVICE LATENCY AND METERING	22
2.4 SUMMARY.....	25

CHAPTER 3 THE IMPACT OF USING A TRUSTED THIRD PARTY IN VEHICULAR CLOUDS 26

3.1 INTRODUCTION	26
3.2 RELATED WORK	27
3.3 PROVISIONING DELAY EFFECT OF PARTAKING A TRUSTED THIRD PARTY	28
3.3.1 SYSTEM MODEL OF PARTAKING A TRUSTED THIRD PARTY	28
3.3.2 VEHICULAR TRUSTED THIRD PARTY (VTTP) OVERALL ARCHITECTURE	29
3.4 PERFORMANCE EVALUATION.....	32
3.4.1 SIMULATION SETTINGS	32
3.4.2 SIMULATION RESULTS AND ANALYSIS.....	34
3.5 SUMMARY.....	39

CHAPTER 4 QUALITY OF EXPERIENCE-BASED PROVISIONING SERVICES IN VEHICULAR CLOUDS 40

4.1 INTRODUCTION	40
4.2 RELATED WORK	41
4.2.1 QoE IN COMMUNICATION ECOSYSTEMS	42
4.2.2 QoE IN MULTIMEDIA SYSTEMS	43
4.2.3 QoE IN HAPTIC APPLICATIONS AND VIRTUAL ENVIRONMENTS.....	44
4.2.4 QoE IN ONLINE GAMING	45
4.2.5 QoE IN VEHICULAR AD-HOC NETWORKS (VANETs).....	46
4.3 QUALITY OF EXPERIENCE (QoE) IN VEHICULAR CLOUDS	47
4.3.1 VEHICULAR PROVISIONING SERVICES QUALITY OF EXPERIENCE (VPSQoE) ..	47
4.3.2 VPSQoE FORMULATION AND RATING	50
4.3.3 VPSQoE-BASED SYSTEM PROCEDURE	52

4.3.4	VPSQoE-BASED SOFTWARE ARCHITECTURE	57
4.4	SIMULATION SETTINGS AND RESULTS	58
4.4.1	SIMULATION SETTINGS	59
4.4.2	MOBILITY MODEL.....	59
4.4.3	SIMULATION RESULTS AND ANALYSIS.....	60
4.5	SUMMARY.....	66
CHAPTER 5 AN AUCTION-DRIVEN MULTI-OBJECTIVE FRAMEWORK FOR PROVISIONING SERVICES IN VEHICULAR CLOUDS		67
5.1	INTRODUCTION	67
5.2	RELATED WORK	69
5.3	QUALITY OF EXPERIENCE (QoE) FRAMEWORK	72
5.4	AUCTION-DRIVEN QUALITY OF EXPERIENCE APPROACH	75
5.4.1	AUCTION-DRIVEN QoE-BASED SOFTWARE ARCHITECTURE	76
5.4.2	AUCTION-DRIVEN QoE PROVISIONING (AQoEP) APPROACH.....	78
5.4.3	AQoEP SELECTION AND JOINING	81
5.5	PERFORMANCE EVALUATION.....	84
5.5.1	SIMULATION SETTINGS	84
5.5.2	SIMULATION RESULTS	85
5.6	SUMMARY.....	89
CHAPTER 6 AN INTERACTION GAME SYSTEM FOR SERVICE PROVISIONING IN VEHICULAR CLOUDS.....		91
6.1	INTRODUCTION	91
6.2	GAME-BASED OF THE QoE FRAMEWORK.....	92
6.2.1	GAME DESCRIPTIONS	93
6.2.2	OBJECTIVES	94
6.3	SYSTEM MODEL.....	94
6.3.1	“GAME N” MODEL	95
6.3.2	MMIGS MODEL	96
6.3.3	“GAME A” MODEL	103

6.4	ANALYSIS OF OUTCOMES UNDER MMIGS	104
6.5	PERFORMANCE EVALUATION.....	108
6.5.1	SIMULATION SETTINGS	108
6.5.2	SIMULATION RESULTS	109
6.6	SUMMARY.....	116
CHAPTER 7 CONCLUSIONS AND FUTURE RESEARCH		117
7.1	CONCLUDING REMARKS	117
7.2	FUTURE RESEARCH.....	118
REFERENCES.....		120
APPENDIX A CLOUD COMPUTING (CC) PARADIGM		132
APPENDIX B CONFIDENCE INTERVALS.....		140

LIST OF FIGURES

Figure 2.1: Evolution technologies of vehicular cloud computing.....	10
Figure 2.2: Taxonomy of vehicular cloud computing applications and services	12
Figure 2.3: Vehicular cloud computing architecture [9].....	14
Figure 2.4: Taxonomy of security and privacy challenges in vehicular clouds.....	20
Figure 3.1: Vehicular trusted third party overall system architecture	30
Figure 3.2: Average service latency in LS - Gas price	35
Figure 3.3: Average service latency in LS - Weather condition.....	36
Figure 3.4: Average service latency in LS - Traffic condition	36
Figure 3.5: Average service latency in HS – Audio streaming.....	37
Figure 3.6: Average service latency in HS – Video streaming.....	38
Figure 4.1: Relationship of QoE and QoS in network model	42
Figure 4.2: QoE framework for provisioning services in vehicular cloud	49
Figure 4.3: Scenario A: System procedure without QoE interest	54
Figure 4.4: Scenario B: System procedure with QoE interest	56
Figure 4.5: VPSQoE-based software architecture	57
Figure 4.6: Performance of average delays - VPSQoE modes – Static	61
Figure 4.7: Performance of average cost - VPSQoE modes – static	62
Figure 4.8: Performance of average revealed information - VPSQoE modes – static	62
Figure 4.9: Performance of average delays - VPSQoE modes – random.....	64
Figure 4.10: Performance of average cost - VPSQoE modes – random.....	65
Figure 4.11: Performance of average revealed information - VPSQoE modes – random	65
Figure 5.1: The QoE framework under study [127]	73
Figure 5.2: Auction-driven QoE-based software architecture	76
Figure 5.3: AQoEP selection procedure scenario	79
Figure 5.4: Example of AQoEP selection and joining.....	82
Figure 5.5: Delay performance	86
Figure 5.6: Service cost performance	87
Figure 5.7: Revealed information performance	88

Figure 5.8: Level of complexity under AQoEP modes.....	89
Figure 6.1: Game-based of the auction framework	93
Figure 6.2: <i>Game n</i> approach.....	95
Figure 6.3: An example scenario of the MMIGS game approach	99
Figure 6.4: QoE (i,j) preference comparison	101
Figure 6.5: QoE (n) proposed offer to vehicle drivers'	101
Figure 6.6: <i>Game a</i> approach.....	104
Figure 6.7: Game n delay.....	110
Figure 6.8: <i>Game n</i> service cost.....	111
Figure 6.9: Game n privacy	112
Figure 6.10: The impact of the QoE-aware models on service delay	112
Figure 6.11: The impact of QoE-aware models on service cost.....	113
Figure 6.12: The impact of the QoE-aware models on privacy.....	114
Figure 6.13: MMIGS-WN average percentage saving	115
Figure 6.14: MMIGS-WN equilibrium.....	115
Figure A.1: Layers of the cloud	133
Figure A.2: Types of the cloud	135
Figure A.3: Mobile cloud computing model.....	136
Figure A.4: Interactions and access of cloud layers	137

LIST OF TABLES

Table 2.1: Summary of open issues in vehicular cloud, potential solutions, and their drawbacks.	24
Table 3.1: VTTP simulation scenarios.....	33
Table 4.1: Static values of QoE coefficients.....	60
Table 4.2: VPSQoE simulation scenarios.....	63
Table 5.1: AQoEP simulation scenarios	84
Table 6.1: Simulation scenarios and corresponding settings	109
Table A.1: Summary of CC, MCC and VCC main features.....	138

LIST OF ACRONYMS

AI	Artificial Intelligence
AQoEP	Auction-driven Quality of Experience Provisioning
CaaS	Cooperation as a Service
CC	Cloud Computing
CI	Confidence Interval
DASH	Dynamic adaptation Streaming over HTTP
DSHR	Dedicated Short Rang Communication
EWMs	Emergency Warning Messages
HC	Hybrid Cloud
HUI	Haptic User Interface
IaaS	Infrastructure as a Service
INaaS	Information as a Service
ITS	Intelligent Transportation Systems
MCC	Mobile Cloud Computing
MMIGS	Multiagent Multiobjective Interaction Game System
MMIGS-NN	Multiagent Multiobjective Interaction Game System No Negotiation
MMIGS-WS	Multiagent Multiobjective Interaction Game System With Negotiation
NaaS	Network as a Service
NQoEP	Naïve Quality of Experience Provisioning
NS	Network Simulator
OSG	Online Social Gaming
PaaS	Platform as a Service
PKI	Public Key Infrastructure
QoE	Quality of Experience
QoS	Quality of Services
SaaS	Software as a Service
SDF	Switching Degration Factor
SLA	Service Level Agreements

SP	Service Provider
STaaS	Storage as a Service
TTP	Trusted Third Party
V2I	Vehicle-to- Infrastructure
V2V	Vehicular-to-Vehicular
VANET	Vehicular Ad-Hoc NETwork
VC	Vehicular Cloud
VCC	Vehicle Cloud Computing
VD	Vehicle Drivers
VN	Vehicular Networks
VPSQoE	Vehicular Provisioning Services Quality of Experience
VTTP	Vehicular Trusted Third Party
VuC	Vehicle using Cloud
WAVE	Wireless Access in Vehicular Environment

LIST OF SYMBOLS

VD_i	Vehicle Driver i
G_i	Game of one level I
G_{ij}	Game of two levels (I and J)
Pr_X	Profit of entity X
Pr_X^m	Marginal profit of entity X
SP_j	Service Provider j
TTP_n	Trusted Third Party n.
$a_1, \dots a_i$	Possible game outcomes of player a
$b_1, \dots b_i$	Possible game outcomes of player b
Ω_P	Set of game preferences
Ω_o	Set of game outcomes
γ	QoE coefficient of privacy
A	Game decision (Accept)
D	Delay measured in time units
I	Privacy measured in revealed information units
P	Price measured in currency units
QoE	Quality of Experience
R	Game decision (Reject)
SC	Service Cost
$SF(X)$	Service Fee payment mad to entity X
i	Interaction level of vehicle drivers
j	Interaction level of service providers
n	Interaction level of trusted third parties
S, s_i	Set of services and service i
α	QoE coefficient of delay
β	QoE coefficient of price

CHAPTER 1 INTRODUCTION

1.1 OVERVIEW

A vital need for wealthier applications and services has emerged over the past few decades, and the new computational paradigm of Cloud Computing (CC) has attracted significant attention. The latest benefits, brought about largely by mobile communication and the extensive use of smart phones, are leading the way to a mobile Internet and providing end users with rich mobile experiences.

CC involves consolidating various existing computing services and resources, based on the pay-as-you-go approach. Basically, a cloud can be considered as an inter-data center network that provides infrastructure, platform and software services [1][2]. This broad definition can be tailored to meet specific requirements, such as on-demand, metered delivery of services in a secure and scalable manner over the Internet [3]. Advances in CC have made it possible to provide software and infrastructure services to mobile users, improvements that address many existing challenges.

Mobile Cloud Computing (MCC) is a new and evolving technology that has attracted interest from both the academic and industry. MCC has been designed to combine the benefits of wireless networks, CC, and mobile computing, and deliver rich computational resources to mobile users and other parties [4]. Similarly, Vehicular Cloud Computing (VCC), an emerging branch of MCC, has inherited the benefits of CC and vehicular communications [5]. Vehicle drivers can use mobile clouds via their mobile devices or in-vehicle computers virtually anywhere, and process any type of on-demand services at any time [6][7].

Recently, VCC has been proposed as a promising solution for vehicular networks challenges, and VCC technologies and applications have been deployed to support the Intelligent Transportation Systems (ITS) paradigm [8]. VCC can be defined as a group of vehicles using corporate hybrid technology, supported by computing resources, storage, communications and access to the Internet that have a remarkable impact on traffic

management, road safety and on-demand provisioning services. There are numerous ITS applications that can benefit from VCC, including traffic management, road monitoring and safety, infotainment related applications, and emergency response systems [9].

Typically, any smart vehicle can act as a mobile storage and/or processing unit, due to the many available on-board resources and services that provide an ideal environment for public service and safety. Although vehicular clouds have a broad range of benefits and advantages for various domains and applications, there are several obstacles that need to be addressed before they can become widely adopted. From a vehicle driver's perspective, privacy, service costs and latency are the most critical challenges [10][11][12].

The security and privacy issues of vehicular cloud systems have not been studied as closely as other potential problems. In addition to the security of the communication medium, the main security and privacy drawback is the lack of control of data stored and/or processed over virtual and distributed resources [13]. Service cost and latency are also concerns for drivers, since they both impact on-demand service provisioning [14][15].

Due to vehicle mobility and the unpredictable topology changes of vehicular networks, it is very challenging to control the three aforementioned aspects simultaneously [16]. A vehicular network is comprised of several vehicles moving at higher speeds than any node in a typical mobile cloud environment, which makes it very difficult for the driver and the service provider to manage the network connections. Frequent topology changes can cause extended delays and increased cost for drivers. Using a reliable, well recognized Trusted Third Party (TTP) to handle the communications between drivers and service providers will resolve these potential contributors to the larger problem.

1.2 MOTIVATION

VCC will likely become a leading technology in the near future, due to its effectiveness and the number of applications and services it can support. One of the most important applications from the driver standpoint is on-demand provisioning services.

Vehicles are becoming more sophisticated and intelligent due to the growing number of on-board applications and resources. They are typically equipped with powerful on-board computing capabilities with access to unlimited storage, as well as powerful communication tools with no limitations on battery life. Consequently, VCC can be used to provide vehicle drivers with on-demand provisioning services at low cost, as well as real-time traffic control services regarding accidents and safety issues [10][17]. However, these developments will raise many issues and challenges.

Service providers have full control of the data in a cloud, which are stored in a virtual infrastructure. Users are unaware of the physical location where services are executed, so the privacy of user data is a significant concern for cloud customers [18][19].

It should be noted that privacy is directly related to the amount of information revealed to a service provider; the more the information revealed, the greater the potential for privacy to be violated [20]. In addition, as yet there are no mechanisms to provide drivers with the service costs of different service providers, or service-price matching. As well as the impact on time efficiency, provisioning latency also increases service usage time, which leads to higher charges for the user [15][21][22].

Due to the privacy issues identified above, drivers are often reluctant to disclose all their personal information. Ensuring the privacy of driver data and personal information while switching between providers is also of concern. Users want to be confident that their information will not be used inappropriately, or sold to a third party when switching to a new service provider [23][24]. Vehicle drivers are likely be concerned about one or more of the abovementioned privacy factors.

We found no studies in the literature that addressed these challenges together, or investigated the effects of one on the other. This needs to be improved by comparing the work done on security and privacy as well as cost and latency of MCC, which seems to maturing.

These challenges and limitations highlight the need for development of a novel framework for provisioning services in vehicular clouds. The framework would manage,

customize and better address drivers' needs for continuously changing requirements, bounded service latency and service cost savings, as well as preserve driver privacy by reducing revealed information. Based on the above, the following three issues must be addressed before VCC can be widely adopted, in order to motivate drivers to join VCC: *service latency, service cost and privacy.*

1.3 OBJECTIVES

The research objectives of this thesis can be summarized as follows:

1. To investigate and review vehicle drivers' requirements, opportunities and challenges in a vehicular cloud environment.
2. To propose and develop a novel framework to provide several provisioning services at low cost with minimal revealed information and provisioning latency, to satisfy the derived requirements.
3. To integrate the concepts of Quality of Experience (QoE) and Trusted Third Party (TTP) within the proposed framework, to achieve system scalability and optimum efficiency.
4. To formulate and test different approaches and schemes against the proposed framework using theoretic approaches to prove the concept, such as auction and game theory. This is expected to be very challenging since latency, cost and privacy contradict each other.
5. To attain the highest achievable performance of the proposed approaches and compare it to the existing conventional models, to ensure minimal latency, fair charges to drivers, and only the necessary amount of revealed information to third parties.
6. To propose and develop a policy-based location-aware mobility framework for provisioning services in vehicular cloud systems.

1.4 THESIS CONTRIBUTIONS

The goal of this thesis is to investigate and combine new paradigms and develop novel approaches for provisioning services in vehicular cloud. Several tools and software applications were used during the design and the evaluation of the proposed approaches, and the research also included many theoretical analyses and experiments. The major contributions of this thesis are as follows:

1. Introduced the concept of utilizing a trusted third party and quality of experience in a vehicular cloud between vehicle drivers and service providers.
2. Proposed a trusted third party framework in different scenarios with various type of services including light and heavy ones.
3. Proposed and developed a novel QoE framework with demonstrated scalability and efficiency that can be used by vehicle drivers and service providers to enhance their mobile interactions. The framework will require that a TTP act on behalf of the drivers and service providers, to ensure the delivery of the services and the quality of the participants' experience.
4. Proposed a generalized multiobjective framework for provisioning services in vehicular clouds.
5. Proposed an auction-driven multi-object model to address and overcome investigated vehicular challenges, including latency, cost and privacy.
6. Proposed a multi-agent and multi-objective interaction game system for service provisioning in a vehicular cloud. Using game approach, the system latency will be bounded and will significantly reduce service costs and preserve privacy, compared to the performance of conventional models.

1.5 THESIS OUTLINE

The thesis is organized as follows:

Chapter 2 presents the state-of-the-art knowledge in the field of mobile cloud and vehicular cloud, and provides an overview of the architectures, applications and approaches used with these technologies. It also highlights the most relevant issues and challenges in vehicular cloud. In *Chapter 3*, we describe the impact of using a trusted third party in vehicular cloud. It also highlights and reviews the most relevant related work and potential contributions in this area. *Chapter 4* presents a novel framework for provisioning services in a vehicular cloud with various providers. *Chapter 5* discusses an auction-driven multi-objective provisioning framework in a vehicular cloud, and proposes an auction theory model with the QoE framework that guarantees maximum savings for system participants. *Chapter 6* presents the development of an interaction game system for service provisioning in vehicular clouds that uses the concept of game theory to guarantee QoE framework participants mutual benefits and fair game treatment. And finally, conclusions and future directions are presented in *Chapter 7*.

1.6 LIST OF PUBLICATIONS

- **Journal Papers**

1. M. Aloqaily, B. Kantarci, H. T. Mouftah, "*Multiagent/Multiobjective Interaction Game System for Service Provisioning in Vehicular Cloud*", *IEEE Access*, vol. 4, pp. 3153-3168, 2016.
2. M. Aloqaily, B. Kantarci, H. T. Mouftah, "*A Policy-based Location-aware Mobility Framework for Provisioning Services in Vehicular Cloud Systems*", (Submitted).

- **Conference Papers**

1. M. Aloqaily, B. Kantarci, H. T. Mouftah, "*An Auction-driven Multi-objective Provisioning Framework in a Vehicular Cloud*", *IEEE GLOBECOM Workshop on Cloud Computing Systems, Networks and Applications*, San Diego, CA, USA, pp. 1-6, Dec., 2015.
2. M. Aloqaily, B. Kantarci, H. T. Mouftah, "*Vehicular Clouds: State of the Art, Challenges and Future Directions*", *IEEE Conference on Applied Electrical Engineering and Computing Technologies (AEECT2015)*, pp. 1-6, Nov., 2015.
3. M. Aloqaily, B. Kantarci, H. T. Mouftah, "*A Generalized Framework for Quality of Experience (QoE)-based Provisioning in a Vehicular Cloud*", *IEEE International Conference on Ubiquitous Wireless Broadband (ICUWB)*, Montreal, QC, Canada, pp. 1-5, Oct., 2015.

4. M. Aloqaily, B. Kantarci, H. T. Mouftah, "*On the impact of Quality of Experience (QoE) in a vehicular cloud with various providers*", IEEE 11th Intl. Conference HONET-PfE, Charlotte, NC, pp. 94-98, Dec., 2014.
5. M. Aloqaily, B. Kantarci, H. T. Mouftah, "*Provisioning Delay Effect of Partaking a Trusted Third Party in a Vehicular Cloud*", IEEE Global Information Infrastructure and Networking Symposium (GIIS), Montreal, QC, Canada, pp. 1-3, Sep., 2014.
6. Aloqaily, M., Kantarci, B., and Mouftah, H. T., (2016) "*Fairness-Aware Game Theoretic Approach for Service Management in Vehicular Clouds*", Submitted to IEEE ICC 2017 SAC Symposium Cloud Communications and Networking Track, 2016.

- **Posters**

1. M. Aloqaily, B. Kantarci, H. T. Mouftah, "*Quality of Experience (QoE) in Vehicular Cloud*", The 5th WiSense Workshop, Ottawa, August 2013.

CHAPTER 2 VEHICULAR CLOUD COMPUTING - STATE OF THE ART

2.1 INTRODUCTION

Individuals and enterprises are drawn to cloud-based solutions for several reasons, as stated in [25]. With the advent of mobile computing and communications, the market share of smartphones has grown significantly, and mobile applications for computing, processing and storage purposes have become an integral aspect of the mobile Internet [26].

Most researchers and developers agree that cloud applications transform how *Information and Communication Technology* (ICT) manages business, without necessarily changing the technology. However, transforming business requires reformation of existing ICT solutions [27]. The main benefits of Cloud Computing (CC) are that it provides rapid, flexible and scalable access to computing resources at any time, based on pay-as-you-go methodology.

The conceptual cloud model is inspired by electricity distribution via the power grid model, so customers do not need electric generators as power is available through normal outlets, and usage is metered and charged accordingly. In addition, if this idea is applied to the ICT business, ICT users will not be required to install software on their local devices. Similarly, enterprise companies will not need to invest more in their IT infrastructure, since they can lease additional infrastructure. This business model consolidates various computing services and storage resources on remote platforms, and the platforms are on designated infrastructures [1][2]. This also highlights the need to stipulate requirements, such as on-demand, metered delivery of the services in a secure and scalable manner over the Internet [3]. For more information about the fundamentals, architectures, and benefits of CC and Mobile Cloud Computing (MCC), please refer to Appendix A.

Cloud computing and vehicular communications are emerging and important fields, and they will contribute to the realization of smart cities. Cloud computing is one stage in the development of vehicular cloud computing, which combines the benefits of mobile cloud computing and vehicular communications. Recent studies indicate that vehicular cloud computing will be the gateway to the future of transportation systems.

A vehicular cloud is a group of moving vehicles that is formed autonomously and can provide many applications and services that benefit vehicle drivers and passengers, pedestrians and city planners, as well as improve emergency response systems. Research communities expect vehicular cloud computing will support computational provisioning services to drivers at appropriate costs. Such services will play an important role in improving the safety of transportation systems, and reduce traffic congestion.

According to the 2013 global status report on road safety [28] and data collected from 182 countries, total road traffic deaths are close to 1.24 million per year, and global road traffic injuries are estimated to be the eighth leading cause of death, similar to the number caused by fatal diseases such as malaria. Due to this and other factors, the need for supportive technology to provide advanced applications and provisioning services related to traffic management, and enable simple and effective access to information, has grown extensively.

Smart vehicles can be defined as an automobile with advanced electronics to support enhanced communications, and provide more storage, computing resources and sensing services. However, the high mobility of such environments, and the early stages of development lead to increased complexity and numerous challenges. Some challenges are related to the technical aspects of development, while others involve adapting the technology to the surrounding environment and authorized parties. In this chapter, we present a literature review to better understand these issues and position them within the scope of the overall challenge, and explain the basic concepts and technologies to help readers understand fundamental vehicular cloud mechanisms. The chapter is divided into three areas: i) cloud computing and mobile cloud computing systems, which basically covers the definitions, architectures and system applications; ii) explanations of the

classifications, definitions, standards and relevant technologies of vehicular cloud computing; and iii) examination of the open issues and remaining challenges of vehicular cloud computing and potential solutions.

2.2 VEHICULAR CLOUD COMPUTING (VCC)

Today, vehicular computing and communications is a popular research area, as vehicles will need improved communications, more on-board services and greater storage. A vehicular cloud can be defined as a group of autonomously formed moving vehicles with the ability to coordinates sensing, computing and communications, and storage to provide a large number of applications and services. Vehicles connected to the cloud can be either Service Providers (SPs) or service consumers. In the context of this research, we are handling the vehicles as a service consumers. Cloud vehicles typically share computing and storage resources via a wireless network backbone [6][9][29]. The most recent evolution technologies of vehicular cloud computing are shown in Figure 2.1. It combines the benefits of vehicular ad-hoc, cloud computing, and mobile cloud computing.

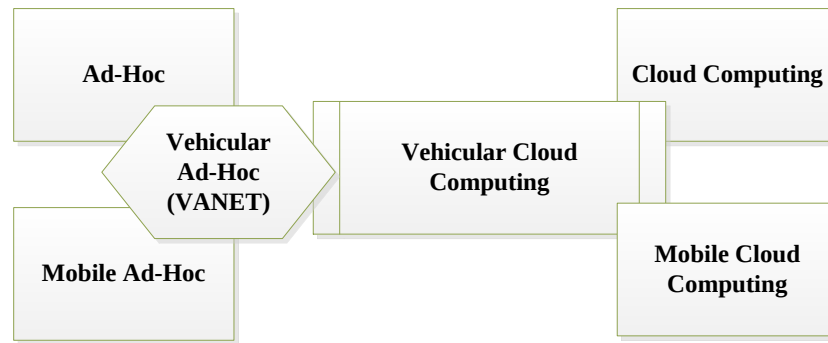


Figure 2.1: Evolution technologies of vehicular cloud computing

The emerging concept of vehicular cloud has been proposed in many recent works [30][31][32]. As vehicles get smarter with supplementary on-board gear, they are capable of handling more complex operations, and unlike other mobile devices, mobile *vehicle* devices can provide real-time functionality, location based services, provisioning services, and storage, with none of the drawbacks of traditional mobile devices. Combining vehicles' resources, and qualifying them to provide cloud services to the

public, has freed visionaries to propose connecting vehicles and turning them into “vehicular” service providers [30].

Allowing vehicles to connect and participate in providing real-time services is enormously valuable, not only for transportation systems but individuals’ lives as well. This is also a great opportunity to make the unused and idle resources of dispersed vehicles available to surrounding parties. Vehicles can trade with or get credit for some of their services with nearby vehicles, passengers or even service providers.

Another advantage of mobile vehicular research was inspired by crowdsourcing and the need to improve drivers’ experience and safety. Recent studies [33][34] have proposed crowdsourcing-based vehicular networking to achieve these. The following sub-sections will overview the taxonomy of vehicular cloud, present fundamentals of vehicular cloud architecture, and discuss its advantages.

2.2.1 TAXONOMY OF VEHICULAR CLOUD COMPUTING

Vehicular cloud computing taxonomy architectures are comprised of multiple cooperating entities. Several applications and services can be useful in the taxonomy, including managing parking lots, accident alerts, monitoring road safety, controlling the evacuation plans of shopping malls and managing traffic lights (Figure 2.2). Vehicular cloud networking service is one of most useful means of sharing information and storage between vehicles. Vehicles with Internet access can provide this service to other vehicles that do not have an Internet connection.

Several taxonomies of vehicular cloud computing have been introduced recently. In [29], the authors classified it by purpose: Vehicular Clouds (VC), Vehicles using Clouds (VuC), and Hybrid Clouds (HC). For example, there could be static or dynamic inter-vehicle rental resources, or vehicles using clouds.

2.2.2 VEHICULAR CLOUD ARCHITECTURE

Since the beginning of vehicular cloud computing technology, research communities have proposed different architectures to address its fundamental components and functions. Any potential architecture requires to have the following communication modules [35]: vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), vehicle-to-broadband cloud (V2B), and intra-vehicle communications. The most comprehensive and relevant architecture found in the literature was presented in [9], as shown in Figure 2.3 [9].

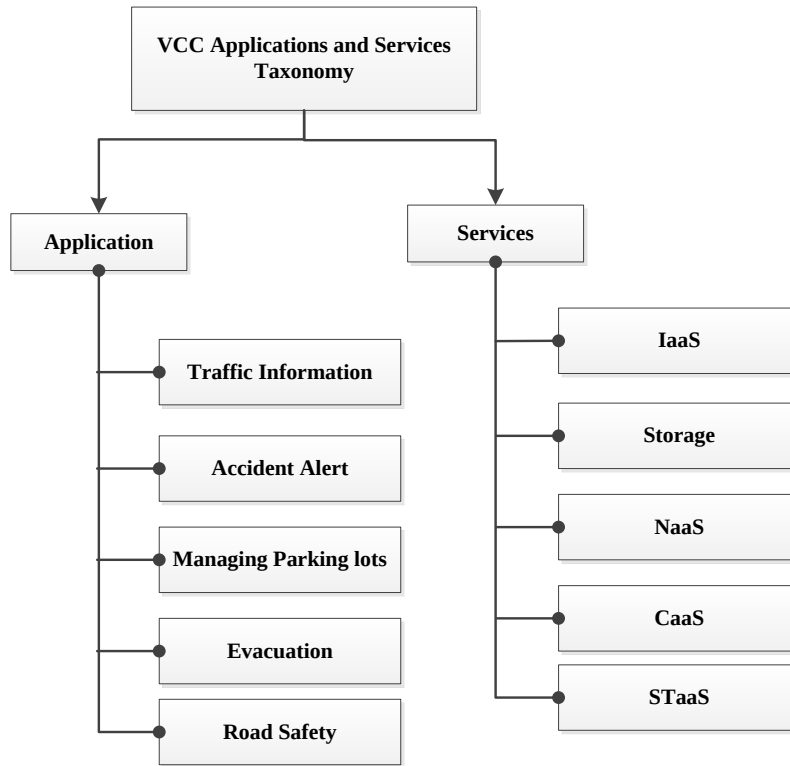


Figure 2.2: Taxonomy of vehicular cloud computing applications and services

VCC architecture essentially consists of three layers: inside-vehicle, communications, and cloud platform. The inside-vehicle layer is responsible for monitoring and collecting

information about the status of the driver in the car, data which can be used at any time to estimate a driver's responses and intentions. This information is usually kept in cloud storage, which is similar to a database unit inside a cloud infrastructure that is part of the communications layer.

As with any networking system, the architecture includes a communication layer. This allows vehicles to exchange information using various communication protocols, such as 3G/4G communications [36], Wireless Access in Vehicular Environment (WAVE) [37] (which is based on IEEE 802.11p), or short range communication protocols [38]. Having such technology available in vehicles will provide flexible communication with surroundings vehicles and service providers.

This architecture includes two communication modules: Vehicular-to-Vehicular (V2V) and Vehicle-to-Infrastructure (V2I). With V2V, vehicles can communicate with each other if they are within range. A V2V sub-layer tracks any abnormal driver behavior, such as exceeding the speed limit or dramatically changing direction, as well as vehicle mechanical problems. If a warning is required, Emergency Warning Messages (EWMs) will be generated which will trigger the car's emergency alarm. The V2I module manages data exchanges between vehicles, infrastructures and the cloud, over wireless networks. This information can be very useful for later study and analysis by governmental or authorized authorities.

Lastly, the cloud platform layer consists of real-time/cloud primary application services, and cloud infrastructure. The application layer makes several applications and services available to the driver remotely at any time, including locations and maps, fuel availability, Network as a Service (NaaS), Storage as a Service (STaaS), Cooperation as a Service (CaaS), and Information as a Service (INaaS), to name few. The cloud infrastructure handles data storage and computation, and all data gathered from vehicle drivers and vehicles will be stored for further study and any later analysis.

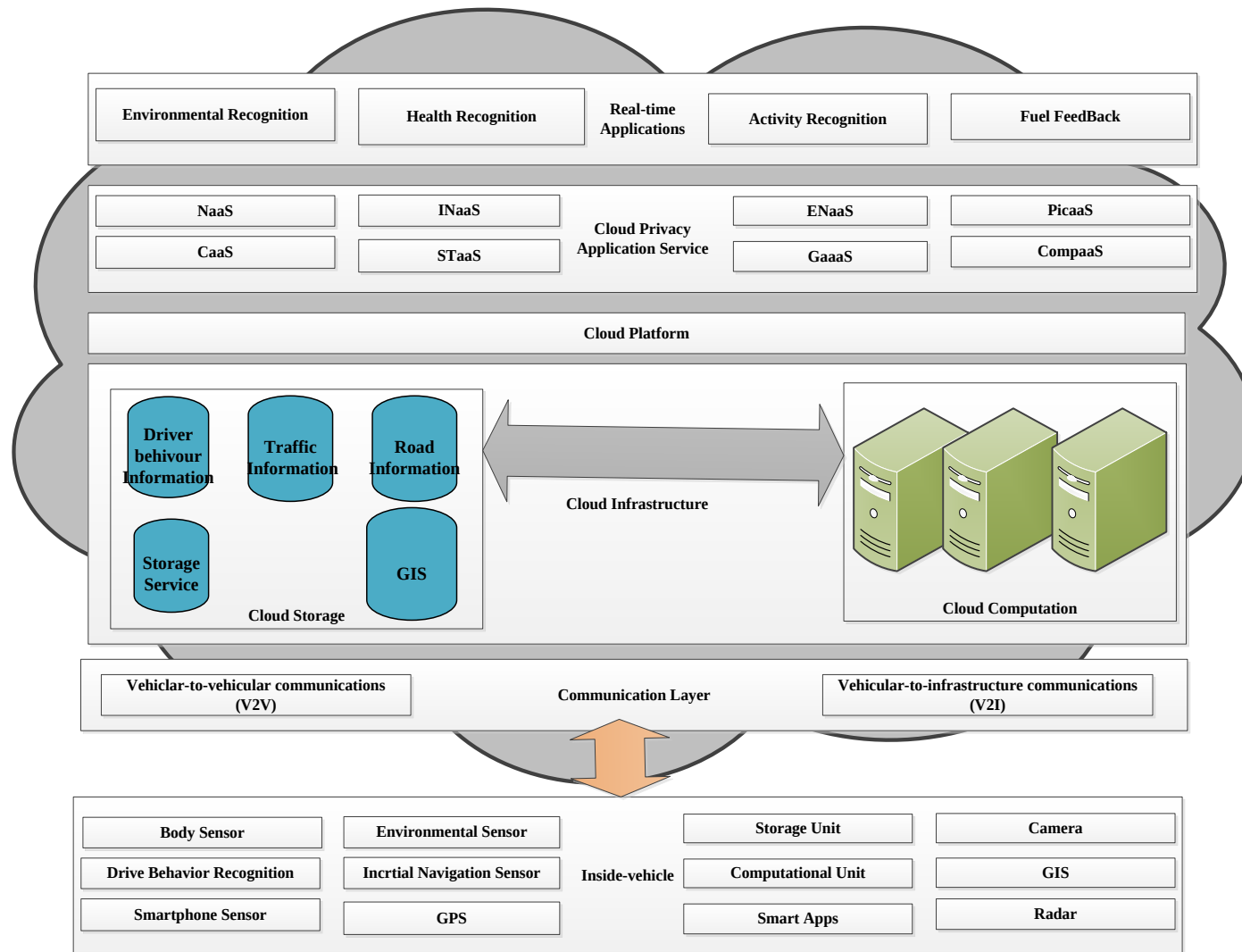


Figure 2.3: Vehicular cloud computing architecture [9]

2.2.3 ADVANTAGES OF VEHICULAR CLOUD COMPUTING

We now know that vehicular networks can be deployed to provide a broad range of benefits to people and organizations, including drivers, transportation systems, passengers, pedestrians, city planners and emergency response systems. They have the flexibility to cooperate with other systems, share services and resources, provide storage and computation and deliver a number of provisioning applications and services. Thus, the wide deployment and correct use of vehicular cloud computing offers many advantages, some of which are:

- Using pay-as-you-go cloud models, which can provide computational services at minimal cost;
- Enhancing ITS applications and services to reduce the number of road accidents and fatalities;
- Providing effective systems for emergency response and evacuation services;
- Allowing vehicles with access to the cloud to be used as service providers that can rent or share capabilities;
- Smart vehicles that can sometimes be more effective than mobile devices, and support on-demand services such as storage, computing, infotainment, accident alerts, parking lot management, road safety monitoring, and traffic lights management;
- A perfect fit for the provision of public sensing;
- Improved driving experience and highly informative vehicular networks; and,
- Access to infrastructure, software, platforms and real-time services.

2.3 OPEN ISSUES AND CHALLENGES IN VEHICULAR CLOUDS AND EXISTING SOLUTIONS

Today, VCC is a promising model for deployment on vehicular networks, though like any new communication technique it does have challenges. There are different VCC architectures and designs in the literature, some of which are more robust than others. Regardless of the architecture, however, there are critical issues and challenges with almost all types of VCC that need to be addressed before it can be widely adopted. In this

section, we briefly discuss these drawbacks, and summarize them and their potential solutions in Table 2.1.

2.3.1 SECURITY AND PRIVACY

Using virtual resources in a cloud, such as storing personal information and data, leads to enhanced control of the data [13][39]. Such enhancement also led to freeing control over such data. VCC combines the benefits of cloud computing and vehicular communications, and thus faces similar challenges. Security is of paramount importance and remains a significant challenge, and drivers' privacy must also be preserved and protected. Drivers should have more control in deciding what personal information can be revealed to others (e.g. service providers, third parties), and what should be kept private [6][40]. In addition, establishing trust relationships between drivers is a vital aspect of reliable communication.

Providing security and privacy in vehicular cloud computing is more difficult than in other networks, due to the high mobility, frequent network topology changes, the dynamicity of the environment and the heterogeneity of vehicles [7]. Several research studies have suggested approaches to resolve security and privacy issues in VCC, including identity management and authentication, personal data vaults, virtualization and scheduling and encryption techniques.

The researchers in [13][40] [41] and [42] presented the privacy and security challenges arising from cloud computing. This research provided promising information to help address these issues, and it is applicable to extensions and future directions. However, the scope of these studies was limited to privacy and security, and did not address the price or delay or their effect on each other. In addition, the abovementioned privacy and security studies are essentially related to typical cloud structure, and do not consider the special requirements of vehicular cloud structures.

The personal data vault solution to protect cloud stored data and limit access to only the owner has been proposed [43]. This method assumes that the vehicle owners can individually control data repositories, which is not possible with some users.

Safeguarding the identity of users, and their personal activities such as location and interest, is also highly important. The authors in [44] proposed the concept of pseudonymity, to hide the actual identities of the users and reduce the possibility of identifying them from their sent data. Other studies [45][46] proposed approaches based on encrypting users' data and searching techniques. A public key encryption and keyword search on encrypted data schemes has also been presented. All this research has yet to address several drawbacks, such as delay, complexity and computation overhead.

Other research [18][23][47] suggests solutions based on identity management to target several cloud computing privacy issues, including authentication, identities, scalability and the complexity of establishing trust relationships between cloud subscribers. These approaches are based on using predicates over encrypted data and multiparty computing, and are prone to the drawbacks of the overhead produced by multiparty computation. Moreover, the deployment of an active bundle in a cloud has been found to be impractical due to the signaling overhead, which requires extensive data storage.

Most of today's mobile applications offer free access to services in return for users' personal information, through a registration process. User mobile personal information, including location, images, contacts and data, are vulnerable in such interactions. Availability, flexibility, and free storage convince mobile users to subscribe to these providers. The authors in [48][49] and [50] used game theory as a tool to determine and interpret stakeholder interactions with user data. An online bookstore scenario was presented, with the user subscribing to access the website services. Several cases have been tested with true and false information. Game theory has been used to analyze the interactions of users, and it improves the quality of service in the cloud. Many proposals that target mobile cloud environments are still in the development stage, and these can be tailored to suit vehicular cloud platforms. Furthermore, the solution models in these studies can be enhanced by incorporating price and latency, though this enhancement is likely to be at the expense of their performance.

Other research studies propose to secure mobile user's data before sending it to the cloud by using encryption, as there are a considerable number of simple and complex

encryption methods. The studies in [45][46] and [51] propose approaches based on encrypting user data and searching scheme techniques, and a public key encryption and keyword search on an encrypted data scheme is presented. However, all this research is encountering several obstacles, such as delay and computation overhead. Some of these studies need to measure the total delay caused by the searching and encryption process, and our aim is to find a simple scheme that is also effective. Such schemes should not cause too high latency, regardless of what level of privacy they provide. These approaches can be extended by mitigating the delay overhead introduced by the encryption-decryption processes. In addition, they are only to support privacy and security on the stored data; personal user information has not been considered.

As we stated, from the work that has been done, there are no existing studies related to cloud computing or mobile cloud computing that can be extended for use with vehicular clouds. Thus, research communities are starting to create and build independent solutions for vehicular cloud issues. The studies in [6][9][24] and [52] focus on vehicular clouds, and introduce the serious security and privacy challenges. They first identified the privacy and security challenges in the mobile environment, and the challenges that are specific to vehicular networks. Vehicular network characteristics, such as high mobility, availability, low error rate and distribution, add additional challenges to some privacy and security methods. For example, relying on probabilistic schemes to provide security in on-demand applications is unacceptable, since even the smallest error could lead to a life-or-death situation in vehicular applications [52].

The authors in [24] introduced new sets of safety and confidentiality messages, data isolation and sanitization, digital signatures and validation of user identity in vehicular clouds. In their study, the proposed sets of messages are used to protect inter-vehicle privacy, while privacy between the service providers and the drivers is not the main concern. Similarly, authors in [52] proposed using a set of primitives to secure vehicular applications.

Olariu and his peers were among the first researchers with a vision of VCC [30][31][53], and they identified possible risks and challenges. In their recent work on VCC security

challenges [18], they focused on vehicular network issues such as how to authenticate high mobility in vehicles, how to create trust relationships between vehicles, and others. Consequently, a security scheme that addresses several of the challenges was proposed. The scheme does not tackle privacy from the driver perspective, and they did not link it to service cost or service latency and study their impact on each other.

The authors in [29] proposed splitting a Vehicular Ad-hoc NETWORK (VANET) into three frameworks: Vehicular Clouds (VC), Vehicles using Clouds (VuC) and Hybrid Vehicular Clouds (HVC). Though the idea of splitting VANET clouds into three frameworks and handling each framework's problems separately seems promising, it is not feasible and has not been implemented. Some argue that problems can arise across communication layers, and require understanding the components of the entire architecture. The same authors also introduced a privacy-aware pseudonymless strategy in VANET [39], in which they outlined the seriousness of vehicle profile generation with identity information. They proposed using a pseudonymless strategy to avoid profiling. The system enhances computational efficiency and bandwidth consumption compared to other systems.

Raya et al. [54][55][56] identified the privacy issue, and proposed a pseudonym system using anonymous public keys and the Public Key Infrastructure (PKI). The public key certification is a long process, and it is frequently updated and can be very complicated. They proposed two possibilities for the certification: by governmental transportation authorities or by vehicle manufacturers. Though this solution does increase the communications overhead, it is prone to huge computation overhead, as well as planning and logistical issues.

To avoid the certification problem, an ID-based cryptography framework was proposed to provide security aspects such as authentication, non-repudiation and pseudonymity, and to handle problems with VANET [57]. However, this framework has more signaling overhead than normal, so no development has been done. In addition, the framework is highly dependent on the infrastructure for short-lived pseudonyms. Similarly, a privacy-preserving defense technique for network authorities to handle VANET problems was

proposed in [58]. This framework employs an identity-based cryptosystem where certificates are eliminated and there is no consideration of service latency or service cost.

Based on the aforementioned studies, we can conclude that security and privacy in a vehicular cloud environment is challenging, and requires serious development and testing effort. The most important requirements of a secure and reliable vehicular system are: a high degree of privacy, certification and confidentiality, authenticity, availability and credibility. Finally, we present Figure 2.4, which is a taxonomy of the different aspects of security and privacy challenges in a vehicular cloud.

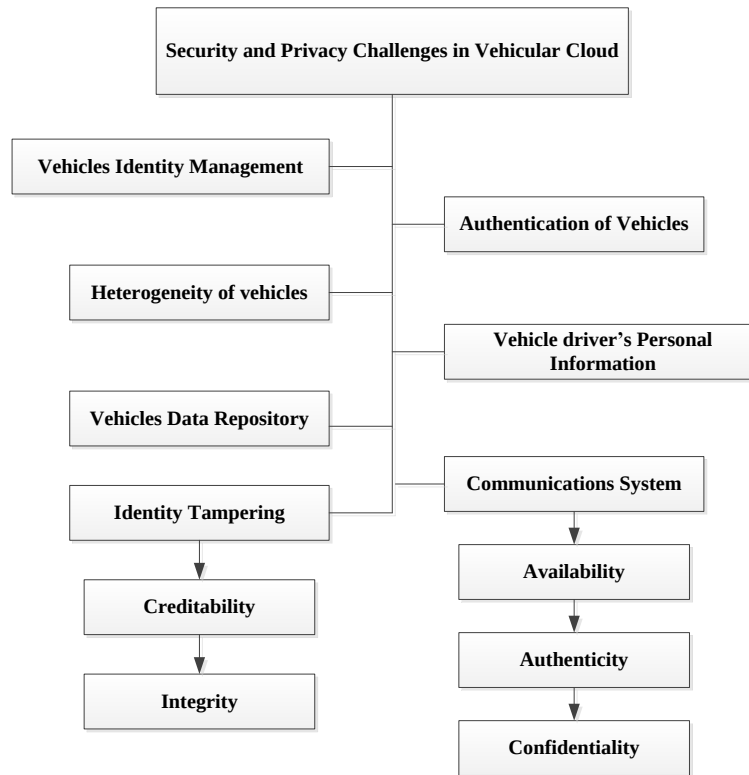


Figure 2.4: Taxonomy of security and privacy challenges in vehicular clouds.

2.3.2 INFORMATION REDUNDANCY

Other issues with VCC are related to the redundancy of information, and idle on-board equipment. A greater number of vehicles on the road will increase the number of resources, which could lead to significant redundant information, and waste a large number of available on-board resources that are idle. The researchers in [59] proposed to

eliminate redundancy by adopting aggregation and clustering methods. A cluster head controls and organizes inter-cluster data and communicates with other cluster heads, but this solution is highly dependable on the type of road. The fundamentality of aggregation also faces security challenges, such as insider attackers who can not only alter their own data but also modify aggregated information. As a result, drivers could base their decisions based on dangerous information. A recent solution [60] suggests a different approach, namely using a resilient aggregation mechanism that leverages existing communication redundancy, then applies a data consistency technique to identify and filter false aggregate information.

2.3.3 IDLE ON-BOARD EQUIPMENT

More vehicles mean there will be more on-board equipment that is not being used. Instead of standing idle, this equipment could be employed to provide applications and services to the surrounding environment, though accomplishing this will require highly effective resource management. Sharing, cooperating or renting applications can be a solution to the resource usage issue, since vehicle equipment will be idle much of the time; at night or when parked for example. Using on-board resources to improve the transportation system and assist drivers, passengers and pedestrians would be an incentive for vehicle owners, since they would be compensated for the service. Compensation would be based on the user and the amount of usage; examples include free parking, tickets, passes or redeemable points [61][62].

2.3.4 INTEROPERABILITY

Cooperation and sharing between vehicles and vehicular providers is another issue related to communications. This is currently in its early stages [35], and there are no rules or regulations for VCC in an interoperable system. Addressing cooperation and sharing between vehicles will require new policies, incentive establishment, control structure and operational management.

Researchers have highlighted several benefits of cooperating and sharing between vehicles and vehicular provider's services, including improving driving experience and

more informative vehicular networks. However, there are many challenges when creating and developing a dynamic cooperative vehicular environment. The authors in [63][64] identified two types of incentive techniques to stimulate cooperation in MANETs: reputation-based and credit-based, and proved their feasibility in mobile ad-hoc networks. However, a considerable degree of adjustment is required to adapt these to VCC. An approach [65] for an incentive scheme to stimulate cooperation between vehicles in VANETs uses a convex function to define the incentives, then overviews the impact of the time and distance with different parameters. The work in [66] proposed a more rigorous approach to stimulate cooperation in VANETs, using coalitional game theory to solve the forwarding cooperation problem and stimulate message forwarding. This is promising, but real word implementation is needed to determine the impact of communication failures.

2.3.5 MOBILITY

Due to vehicles' mobility and the unpredictable topology of a vehicular network, it is very difficult to control communications or anticipate network predictions. This issue identified the need for a dynamic routing protocol that can deliver data in a reasonable time, with minimal dropped or lost packets. Although VCC can enhance system reliability and reduce latency, it is very challenging to develop and test dynamic, efficient routing protocols. Today, the most popular solutions for routing in vehicular networks are: multi-cast protocols, geo-cast protocols, and uni-cast protocols. More recently, carry-and-forward protocol techniques have also been considered [67].

2.3.6 SERVICE LATENCY AND METERING

Vehicular clouds are no longer a visionary technology, but now an attainable goal. Smart vehicles are already equipped with functionality that provides drivers with a variety of on-demand provisioning services, such as passenger entertainment. Vehicle drivers and passengers need to be motivated to use such services, and drivers can be very apprehensive about service cost and latency. There are currently no mechanisms to provide drivers with the cost of different service providers, or available service-price

matching. Another important consideration is that in addition to the impact on time efficiency, provisioning latency also increases service usage time, leading to higher charges for drivers.

VCC challenges related to service latency and metering are still unaddressed. While pricing is an important factor in determining cloud service providers, service latency is critical for certain applications and users. Provisioning latency and pricing have not been fully explored for VCC, and they are aspects that vehicle drivers can be very concerned about. Currently, there is no standard method of provisioning services for drivers in a liberalized cloud service market, and no available service-price matching. In addition to its impact on time efficiency provisioning latency also increases service usage time, which leads to higher charges to the user. Clustering and delay-bounded methods are the solutions most proposed at this time [68][69].

A pricing algorithm for resource allocation in cloud computing environments was proposed by the authors in [70]. The main goal of the study was to maximize the social welfare and characterize the optimal solution by maximizing the social utility through a simple pricing scheme: users charged for services rendered. The unaddressed issue in this solution is the focus on fixed per-usage pricing, while not considering future price change or decrease with other providers. This solution also does not consider any vehicular cloud context and its corresponding requirements. Similarly, the authors in [71] developed a computational pricing mechanism for renting cloud resources. They state the system will be efficient and honest and will use auction-style pricing mechanisms, which will enable users to compete fairly for resources and enable service providers to increase their profits.

Another direction of research focuses on the price distribution of services over the cloud, such as [15][72] and [73]. This research has concentrated on awareness of the electricity cost, in order to reduce it by taking into consideration the cost of delays and the actual cost of service. The authors in [73] proposed an algorithm to minimize service provision costs in both reservation and on-demand modes. The algorithm consists of two phases: The first phase is related to correct resources reservation whereas the second considers

the predicted resource demand. The algorithm seems to improve the quality of service (QoS).

Table 2.1: Summary of open issues in vehicular cloud, potential solutions, and their drawbacks.

<i>Challenges</i>	<i>Potential Solutions</i>	<i>Drawbacks</i>
<i>Privacy and Security</i>	• Identity management, • Personal data vaults, • Virtualization, • Encryption techniques.	• Provisioning latency, • Computation overhead, • Individually controlled data.
<i>Redundancy</i>	• Aggregation, • Sorting and prioritizing • Clustering, • Data consistency.	• Huge communication and computation resources, • Can introduce false aggregates, • Possible to insider attackers.
<i>Ideal on-board Equipment's</i>	• Sharing, • Cooperating, • Renting.	• How to generalize policies and rules, • Rich environment for hackers, • How to apply service charges.
<i>Interoperability</i>	• Reputation-based approach, • Credit-based approaches, • Coalitional.	• Designed for MANETs, • Lack of credibility, • No real implementation, • Communication failures.
<i>Dynamic Routing</i>	• Multi-cast protocols, • Geo-cast protocols, • Uni-cast protocols, • Carry-and-forward.	• Reliable broadcast not guaranteed, • Network fragmentation, • Broadcast storm.
<i>Service Latency</i>	• Clustering and hierarchal models, • Delay-bounded.	• Increases service usage time, • Selection cluster-head increases time, • Re-forming clusters incur extra computation.
<i>Service Cost</i>	• Pricing algorithms, • Price distribution, • Scheduling and pricing.	• No available service-price matching, • Complexity, • No guarantees on privacy.

The research in [21][74] proposed dynamic pricing in resource trading among cloud providers. The authors developed an online scheduling and pricing algorithm, assuming a wireless service provider is servicing users. Results demonstrate a tradeoff between the profit and queueing delay, if the system control parameters are appropriately tuned.

Although the economical and computational advantages of dynamic pricing have been explored, the algorithms require development and testing before they can be applied.

2.4 SUMMARY

This chapter presented a general overview of the revolution of vehicular cloud networks, in a manner that most readers can understand. Much research has been conducted into motivating and accessing the benefits of this technology, as well as the many services and applications that can be deployed. As is typically the case for such breakthroughs, there are many challenges and issues that require careful investigation, and we have identified the critical challenges from the driver's standpoint. Table A.1 summarizes the key aspects of CC, MCC, and VCC.

Finally, vehicular cloud computing is a very rich environment for research and development, even though issues and challenges will arise with every deployment and implementation. From the issues and challenges presented above, drivers are most concerned about the aspects related to their physical component; more precisely, service cost, privacy and service latency. Vehicle drivers will not be as concerned about communication problems in the network than they will about how much they will be charged for a specific vehicular service. Similarly, personal information (i.e. privacy) is a huge concern for drivers, so providing increased privacy will motivate them to engage more with the technology. And service latency can be very frustrating for drivers, and have an impact on service charges.

CHAPTER 3 THE IMPACT OF USING A TRUSTED THIRD PARTY IN VEHICULAR CLOUDS

3.1 INTRODUCTION

The Internet of Things (IoT) has introduced significant changes to conventional computing and communication. The concept is to have billions of uniquely identifiable objects ubiquitously interconnected, and accessible through the Internet backbone. Realization of the IoT paradigm requires significant computational and storage offloading, resulting in the development of cloud computing-based architectures and communication models [75].

Mobile cloud computing is an emerging field that is attracting interest from both academia and industry. Advances in cloud computing have made it possible to provide software, platforms and infrastructure as services to mobile users. These improvements overcome many existing mobile challenges, such as storage and power consumption.

With the advances in mobile networking and cloud functionality, vehicular cloud computing has become a viable application in such environments [53][76]. It emerged from mobile cloud computing and vehicular networking concepts, and has inherited the benefits of both concepts. Vehicle drivers can now join mobile clouds via mobile devices or in-vehicle computers from anywhere, and process all types of data on-demand at any time [6].

Vehicular cloud computing can perform a broad set of on-demand applications and services, which makes it highly suitable for future development. Multimedia content delivery, content sharing, infotainment, Internet access and storage are some examples of services achievable through vehicular cloud functionality.

Although vehicular cloud brings a wide range of benefits to various services and applications, there are several issues and challenges that need to be carefully addressed. Providing provisioning services to vehicle drivers is still a bottleneck, and from a driver

standpoint privacy, service cost and service latency are critical. Vehicle drivers may be concerned about one or a combination of these.

The state-of-the-art has several proposals to address these issues. In this chapter, we begin by reviewing previous work related to a vehicle driver's privacy, service cost and service latency. Next, we present a simple architecture based on vehicular trusted third parties, and study the provisioning delay effect of using a third party in vehicular cloud functionality. This architecture recognizes and adapts well with such environments, and helps address service provisioning challenges in vehicular clouds. It has been tested in different scenarios and services including light and heavy one's.

3.2 RELATED WORK

The crucial need for on-demand provisioning services in vehicular cloud computing environments has been well explained. However, every new technology initially has issues, and vehicle drivers are concerned with service metrics. Service cost and latency, and driver privacy with respect to the amount of information revealed to the service providers, are among these concerns. As we have noted from the previous chapter, no studies in the literature have addressed the concern of provisioning services in a vehicular cloud; most current developments are related to cloud computing and the mobile cloud. In terms of service delay, the work done in [77] mainly addresses the service delay effect of using a trusted third party with a vehicular cloud. Provisioning delay data is obtained via drivers' mobile nodes in the network, and reformulated by a weighted combination of the delays. This work will be discussed in detail in the following sections. Moreover, service cost is less explored for vehicular clouds. We believe this is a very crucial issue in this environment, and calls for the development of an interoperable solution.

To the best of our knowledge, this Ph.D. thesis study has been the first attempt to tackle all these issues together and address them in a holistic manner. Therefore, the motivation of this thesis is to provide a framework that addresses all these concerns through a trusted third party and QoE-based methodology. So far, no research other than that has been done during the course of this thesis study has adopted the concept of the QoE with trusted third party and driver feedback. Thus, our goal is to provide vehicle drivers with

services at low cost, with minimal information revealed to service providers, and optimal provisioning latency in one framework.

3.3 PROVISIONING DELAY EFFECT OF PARTAKING A TRUSTED THIRD PARTY

Several aspects of vehicular cloud provisioning services are of concern to drivers. Service latency, which can be a significant problem for a vehicle driver en route, can be defined as the total delay of the requested service from the vehicle driver to the service provider, as well as on the return. Provisioning services usually have unexpected interruptions that increase provisioning latency and service usage time, leading to higher charges for the driver [77]. In this chapter, we propose a framework to minimize provisioning latency as much as possible by using a Trusted Third Party (TTP). TTPs formulate the requirements and driver needs based on the experience of previous drivers in the cloud. The goal of the proposed framework is to reduce provisioning service latency between vehicle drivers and service providers. In the following sub-sections, we present the proposed system model in detail, discuss the simulation results, and compare the model to other provisioning modes.

3.3.1 SYSTEM MODEL OF PARTAKING A TRUSTED THIRD PARTY

With the growth of vehicular applications and provisioning services, VCC is expected to be the best fit compromise between drivers' needs and service providers' benefits. A service provider's objective is to maximize their income, while drivers want to minimize their expenses. Service providers are expected to provide a certain level of quality of service to vehicle drivers, and minimize provisioning latency. Provisioning service latency can be disturbing for drivers and passengers en route, and could lead to extra cost for the driver. Unjustified extra charges to vehicular subscribers may hamper the successful expansion of the technologies. Communication interruptions and other factors may cause latency, and service providers typically do not consider or compensate the subscribers for such delays. Furthermore, vehicle drivers cannot control or even know if additional charges have been applied to their account due to service latency, and this highlighted the need for a specialized vehicular third party to help of resolve the issue.

Vehicular cloud third parties are professional, specialized organizations that evaluate the communication aspects of vehicular cloud provisioning services, and guarantee the quality of the services between service providers and drivers.

Today, the concept of hiring a third body to act between and provide different parties with professional services has been implemented and tested with many different models; the financial service PayPal is a well-known example of this. Third parties usually provide their publicly available (i.e., free) services to subscribers, though special services and privileges may involve fees. Here, we are not concerned with why third parties are providing publicly available services, though we assume that the TTPs will receive compensation for their services and elapsed time. Indeed, deployment of a TTP may be at the expense of additional delay. In this chapter, we study the impact of a trusted third party acting on behalf of vehicle drivers and service providers to deliver provisioning services with minimum delay. The main question here is: Is it advantageous to use a vehicular cloud third party or not? The cost of a TTP will be examined in the next chapter.

3.3.2 VEHICULAR TRUSTED THIRD PARTY (VTTP) OVERALL ARCHITECTURE

Figure 3.1 illustrates the overall system architecture of using a trusted third party for vehicular provisioning services. The architecture employs a TTP between the users (i.e. vehicle drivers) and the cloud (i.e. service providers), and consists of three main communication levels: (1) driver level; (2) TTP level; and (3) service provider level.

The architecture shown in Figure 3.1 is also applicable for multi-objective and multi-purpose vehicular use. As shown, different type of vehicle users can be associated with suitable vehicular third parties to meet their specific need. For example, certain vehicular third parties might specialize in municipal services, and a sub-group of this category could leverage third party services such as police and first responders. At this point, we expect regular vehicle drivers who are looking for on-demand provisioning services, such as entertainment and traffic related applications, will be the major users of this proposal. However, vehicular services and applications are expected to grow extensively with the

continued advancements in vehicular technology. Some emerging business and corporation technologies, such as Uber (an online transportation network company), car rental companies and taxi services, are expected to introduce vehicular services to their customers soon in order to increase user satisfaction and compete in the market.

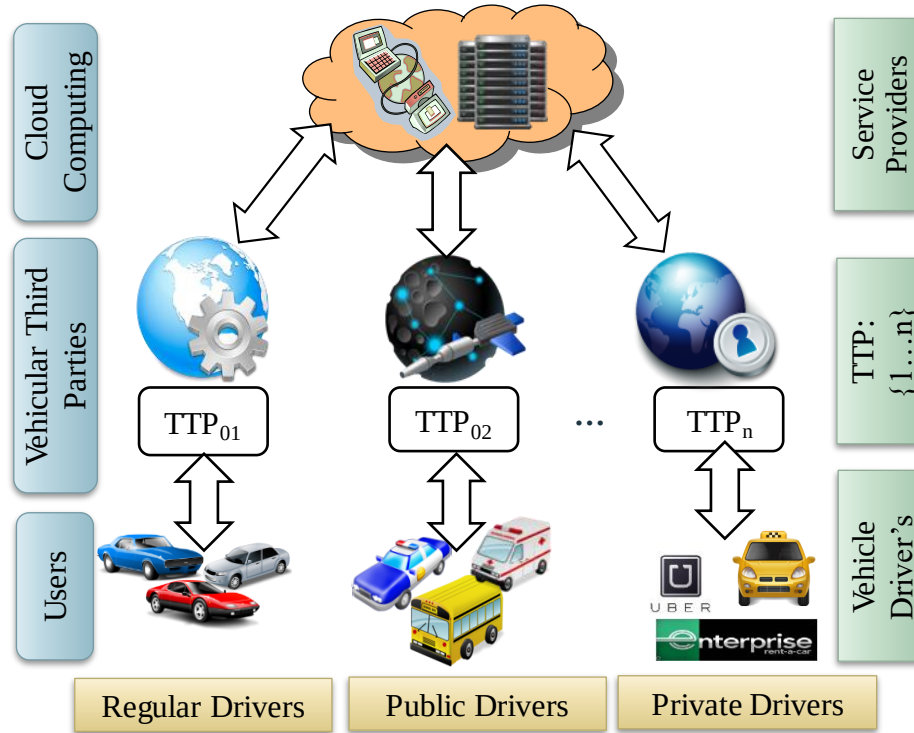


Figure 3.1: Vehicular trusted third party overall system architecture

The trusted third party is the key entity of the proposed architecture, which is responsible for handling all communication aspects with the other two parties. The proposed vehicular TTPs in this architecture are well known, successful commercial organizations that provide and sell services to vehicular users. The position of the trusted third party is mid-way between vehicle driver's and service providers. However, the trusted third party actual location is in the cloud. There are several third parties in practice today, most of which are not suitable for our architecture. There are issues need to be addressed before these existing third parties can be adopted; they are specialized to manage online payment services only, and interoperability and standard interfaces between third parties is still an open issue. The TTPs in the proposed architecture are highly specialized, and

will include most provisioning services that vehicle drivers require. In addition, they will guarantee that the service providers' performance fully meets drivers' expectations. Furthermore, there will be lines of communication among the TTPs that will guarantee consistent quality of service.

VTTP architecture development and communications methods are based on a decentralized approach. Our hierarchical and clustered network setup aims at the following three objectives: 1) Scalability, 2) reduced delay, and 3) minimum computation overhead. As shown in Figure 3.1, the relationship between vehicle drivers and their selected TTPs is formed within one cluster, with the TTPs being the cluster-heads and the drivers the cluster participants that are connected directly to their cluster-head. Thus, the relationship between TTP and service providers is cluster-based. The selection procedure of the cluster participants is based on their interest; they will choose to engage with the TTP that best meets the standard of service they are looking for. In the following chapters, we explore the possibility of dynamic cluster selection to maximize probable revenue in detail. At this level, we limit the focus of this study to the latency aspect.

Another advantage of adopting TTPs for vehicular provisioning services is the negotiation leverage that third parties can bring to bear on service providers. They can communicate and prepare future contracts with service providers to provide certain services to potential drivers, anticipate the statistics of future clients and their most needed services, and plan ahead to deliver the services effectively. These conditions cannot be met individually without specialized TTPs. In other words, drivers cannot purchase a collection of provisioning services for their future use, as it would not be feasible or practical, and would contradict the provisioning on-demand concept. An additional critical consideration is that vehicle drivers will likely use multiple service providers, and it would be more reliable to deal with one well-known professional TTP than various service providers.

In traditional systems, vehicle drivers request their services directly from the first available service provider who can deliver the desired service (Neutral approach). The service providers provide the service, then they charge the driver based on their usage in

terms of data and time. With our proposal, drivers initially join the TTP that best corresponds to their category of interest. They request a service from the TTP, who finds the service from the most suitable service provider at the best price and delivers it to the driver, after which the TTP applies the total charges. Drivers only deal with a cluster-head (i.e. TTP), and are totally unaware of the pool of service providers. If we compare the communication aspects in both models, there is not much change with respect to the drivers; they still only deal with one entity at a time. However, the service delivery from the provider to the drivers will go through a TTP, which is not the case in the traditional models. Nonetheless, TTPs are taking the initiative to prevent or reduce service latency, using their huge infrastructures, early planning and preparation, and experience in provisioning services to manage it. In the upcoming section, we examine the effect of delivery delays to determine how much latency might be expected with such proposal.

3.4 PERFORMANCE EVALUATION

It is important to test and evaluate the performance of any proposed system model, to highlight problems that might arise due the development. Simulation techniques are a good way to achieve this, and network simulators (NS) can be the most appropriate tools in this area. To simulate our proposed architecture we used the Network Simulator 2 (NS-2) [78], a discrete event simulator running at the networking level that can provide results closest to real-world expectations. In the following sub-sections, we present the numerical results of the proposed VTTP architecture with respect to provisioning service latency. We introduced and evaluated two modes for service latency in a vehicular cloud: Service Latency Sensitive Mode (SLSM) and Neutral mode. Our main objective for this evaluation was to compare both modes to determine the impact of using a VTTP on service latency, for both drivers and VTTP.

3.4.1 SIMULATION SETTINGS

Over the past decade, NS-2 has become popular for educational and industrial use. It provides various benefits, including supporting TCP simulation, the capability to simulate simple and complex networking scenarios, and the potential to be deployed in wired and wireless networks. The NS-2 simulator is written in a robust programming language

(C++), which makes it easy to configure and allows it to run quickly. However, it only supports two communication methods (bi-directional and omnidirectional [79], which means the mobility feature must be configured manually. This shortcoming makes it difficult for the nodes to find other moving nodes dynamically. NS-3, which was recently released, has added values and features over NS-2, such as leveraging the use of a python scripting interface to improve scalability and modularity. At this research level we believe NS-2 will meet our objectives, though we consider using NS-3 to evaluate an extension of proposed architectures in the upcoming chapters.

Table 3.1: VTTP simulation scenarios

<i>Scenarios</i>	<i>Vehicles</i>	<i>TTPs</i>	<i>Service Providers</i>
1	5	2, 3	3, 4
2	10	3, 5, 6	5, 7, 7
3	15	5, 9, 9	9, 12, 13
4	20	9, 11, 15	10, 13, 17
5	25	14, 16, 19	18, 20, 22

The simulation communication medium employs Destination-Sequenced Distance-Vector Routing (DSDV) protocol on an IEEE 802.11.p communication stack with 1024 byte/packets. In the following sets of simulation scenarios, each was repeated 10 times and the 95% confidence interval is reported, Appendix A presents more details on confidence intervals computation. Two main modes were evaluated (i.e. *SLSM* and *Neutral mode*) with a range of 5 to 25 vehicles. The simulations will be used to evaluate the possible impact of different scenarios on service latency, and to study the impact of employing the TTP concept in vehicular clouds. In order to assess the behaviour of the proposed architecture in different situations, we simulated multiple scenarios with a random number of vehicular participants. Table 3.1 shows simulation scenarios.

In the simulation scenarios, we randomly selected ranges of 3 to 22 service providers, 2 to 19 TTPs, and 5 to 25 vehicles. We assumed that the vehicles are arbitrarily connected to the TTPs at the start of the simulation, and that the TTPs are aware of all service

providers and their services, and can communicate with them. Many studies in the literature [80][81][82] have setup their simulations with random number of vehicles and monitored the effect of such randomization on the behaviour of their systems. They have shown fair results when it comes to the random number of vehicles, however, the authors in [82] state that the random placement of the vehicles may cause in some cases that nodes being unable to communicate properly with other nodes.

Two different service latency modes were proposed and evaluated for testing and comparison purposes: *SLSM*, which is sensitive to service latency and based on the VTTP architecture proposed in 3.3.2, and *Neutral mode*, which is similar to today's provisioning services models. There was no consideration of specific aspects of the service providers; users simply connected to the first available one to request service(s). In this mode, service providers may not be able to provide the requested services at all times, as they might be rented or out of stock at the time of the request. In similar situations, service providers without knowledge of the client try to acquire the requested service from other providers, which is fine but increases service latency.

3.4.2 SIMULATION RESULTS AND ANALYSIS

We evaluated the abovementioned proposed modes in terms of the provisioning service latency metric. The delay is the end-to-end time from when a driver sends a request to the service provider, to when they start receiving the requested service. We introduced two types of provisioning services in two groups: Light Services (LS): *Gas price*, *Traffic conditions* and *Weather conditions*, and Heavy Services (HS): *Audio streaming* and *Video streaming*. Depending on the nature of the service, the providers are expected to update vehicle drivers with the status of the service with a realistic methodology. In the proposed LS mode, Gas prices and Weather conditions services are delivered to the vehicle drivers once per request since the status of such services will not change over a small period of time. However, some services, like traffic conditions, require more frequent updates than other services. Thus, in the case of Traffic condition service, VTTP is required to deliver three different updates to the vehicle driver within thirty minutes' window. Traffic conditions values provided from providers to VTTP have been classified in the

simulation into three different slots: Normal, moderate, and rush. Normal traffic at the start of simulation, rush traffic at the middle of simulation, and moderate at the end of the simulation. The three different updates on the traffic condition expect to fall with the different traffic slots. When it comes to heavy services settings, it is far more complicated such as video streaming. To better understand our simulation and avoid complexity, we assumed that the driver wanted three minutes of video (~20 Mbytes) and seven minutes of audio (~5 Mbytes).

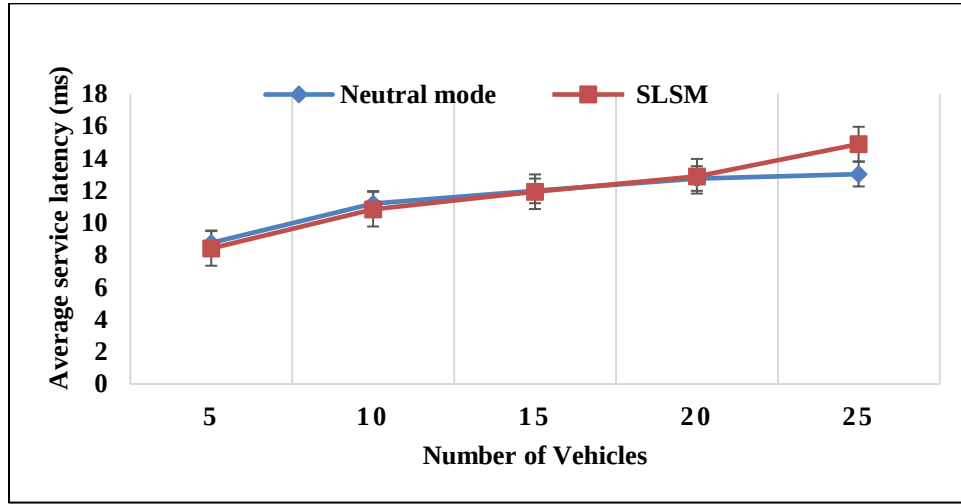


Figure 3.2: Average service latency in LS - Gas price

In the first group, we tested the average delay of the *SLSM* when provisioning LS-type (i.e., Gas price) service and compared it to the *Neutral mode* delay, as shown in Figure 3.2. For such a light service, *SLSM* had delays similar to those of the *Neutral mode*, though the delay increased as more vehicles were involved and was worst with the group of 25 vehicles. Similarly, in the Weather condition set (Figure 3.3) there was no significant delay difference between the two modes. However, as shown in Figure 3.2 and Figure 3.3, *SLSM* had improved the average service latency in low density conditions (~5 to 15 vehicles) then it increased slowly.

In the third set of the first group (i.e. LS – Traffic condition), Figure 3.4 presents the delay performance for the two modes discussed earlier, and shows that *Neutral mode* introduces the least delay compared to *SLSM* delay. This was expected, due to the fact that VTTP updates the service status of the Traffic condition periodically every thirty

minutes, which is not the case in the Neutral mode. This increased provisioning latency cannot be avoided, due to the additional processing time required from the VTTP to the service provider and back to the vehicle driver every thirty minutes. The measured total average delay between the two modes did not exceed 0.7%, which is a minor concern if we factor in the benefits of adopting VTTP and the extra features of SLSM.

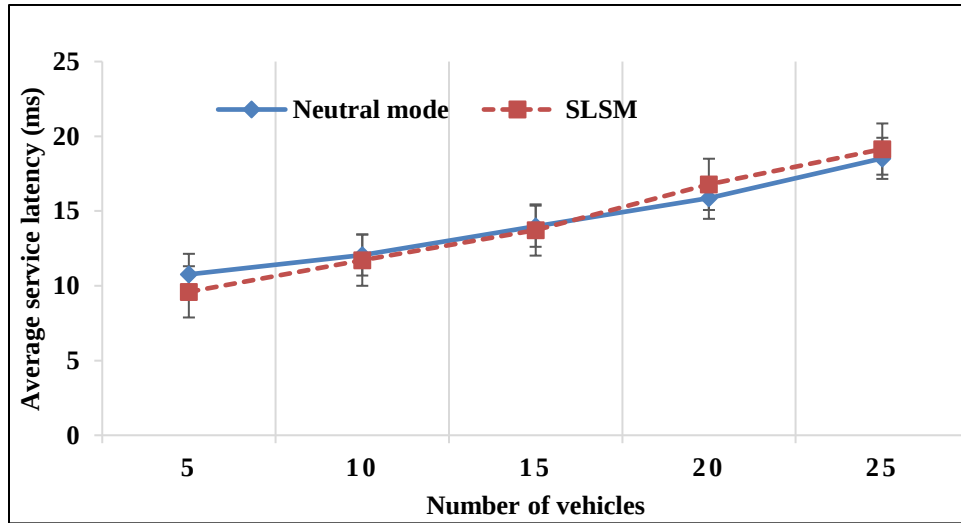


Figure 3.3: Average service latency in LS - Weather condition

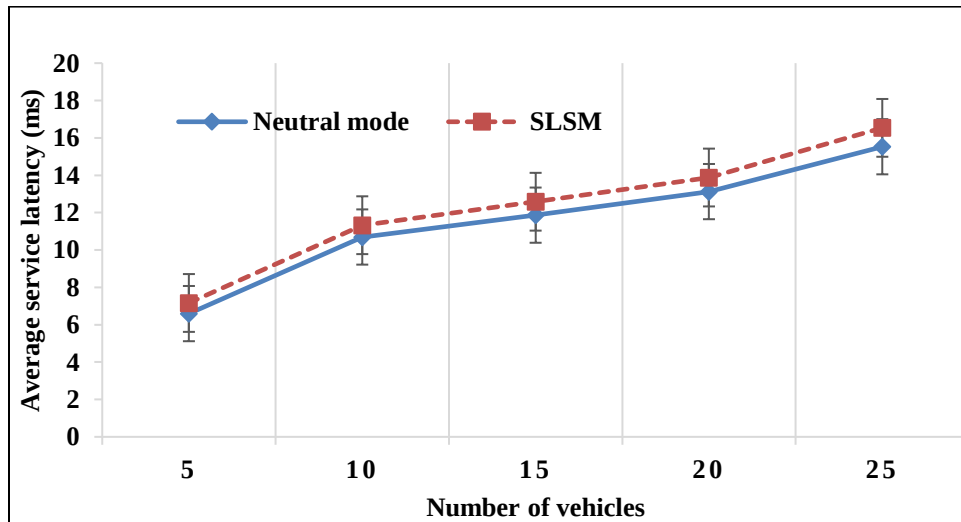


Figure 3.4: Average service latency in LS - Traffic condition

In the second group of evaluations (i.e. HS – Video and Audio streaming), each driver wants to watch the video twice and listen to the audio three times. The TTP in the

proposed VTTP architecture has pre-knowledge about which service provider has the requested material available to stream, information that regular vehicular clients typically do not have access to. This will reduce the provisioning service latency, as shown in Figure 3.5 and Figure 3.6.

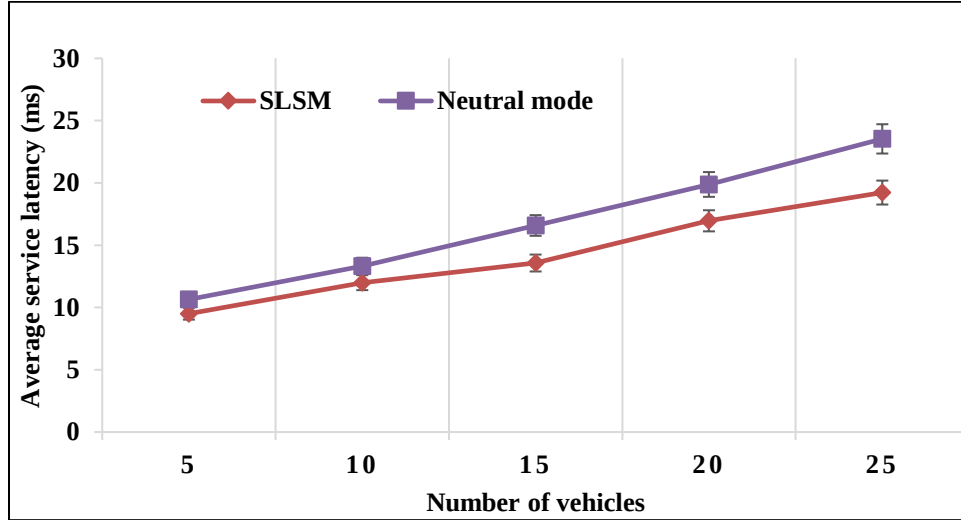


Figure 3.5: Average service latency in HS – Audio streaming

Figure 3.5 compares the average service latency of the SLSM and Neutral modes to stream an audio file distributed over 25 vehicles. It shows that SLSM outperforms Neutral mode, and always provides a lower average service latency. These results also illustrate that with a smaller network density of 5 to 10 vehicles there is little difference between the SLSM and Neutral modes, while increasing the network density to 15 to 25 vehicles results in noticeable improvement. Thus, adopting a third party can help reduce potential provisioning service latency.

Similarly, in the case of video streaming (Figure 3.6) we found that in the second set of the evaluation SLSM outperforms Neutral mode at a very early stage. This is due to the fact that VTTP is aware of all service providers and their available services, thus, VTTP is faster to choose the proper service provider than vehicle driver. The improvements increased gradually as the network density increased, with the best result of 6.3 ms smaller delay registered at 25 vehicles. This value is the average cumulative delays from all delays at dense of 25 vehicles. The second evaluation group (i.e. Heavy service) showed that the type of service and the volume of streamed data can be a factor in

increasing provisioning service latency. Neutral mode failed to maintain less or similar service latency compared to the other mode, and presented a much better result when the number of vehicles increased.

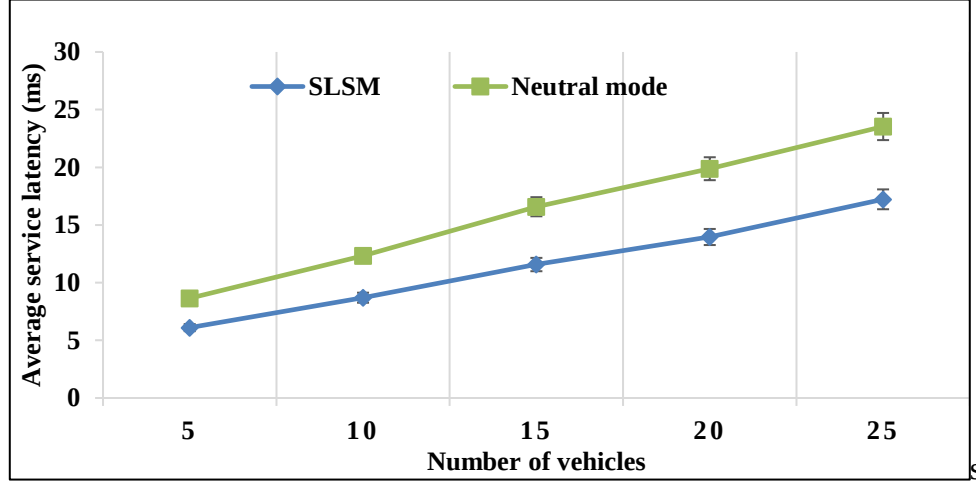


Figure 3.6: Average service latency in HS – Video streaming

Our later results indicate that the network density and type of service play major roles in the provisioning service latency. In addition, using a vehicular trusted third party can significantly help drivers with service provisioning in general, and reduce service latency as discussed earlier. Hence, from a user point of view it is most practical to adopt SLSM-based vehicular trusted third parties for provisioning services in a vehicular cloud. However, this architecture addresses the issue of service latency without considering privacy or service costs, so to be successful it must be generalized and presented to the driver as a complete package. As discussed in 3.2, the three main driver concerns (service latency, cost, and privacy) must also be addressed in this architecture, and the effect of these parameters on each other has to be investigated and identified in a correlated manner. Furthermore, the relationships between the system entities (i.e. drivers, TTPs and providers) require further study, and those between cluster-heads and the participants have to be optimized and employed within a VTTP architecture.

3.5 SUMMARY

Since VCC is an emerging branch of MCC and inherits the benefits of CC and vehicular communications, it faces similar and extended challenges. There are many proposals in the literature addressing the issues arising from CC and MCC, and perhaps some of these models can be extended or modified to resolve VCC issues. However, this is not the case in current studies.

Vehicular cloud provides a number of applications and services to vehicle drivers. On-demand provisioning services is one of the most important services for drivers. This capability is essential since, among other things, it improves the quality of life and driving experience, makes networks more informative, saves lives, and makes access to services and information faster and easier. However, providing provisioning services to drivers is still a bottleneck and faces many challenges. For drivers, privacy, service cost and service latency are the most critical challenges to be addressed.

Motivated by the rise of vehicular provisioning services and the need to resolve the inherent issues and challenges, we proposed a vehicular trusted third party (VTTP) architecture that employs vehicular third parties to provide provisioning services to drivers. The architecture has been designed for multi-objective and multi-purpose vehicular uses, and our objective in this chapter was to solve the challenge of provisioning service latency by using VTTP. Though the proposed architecture has shown promising results compared to other conventional models, in its current state it does not meet and satisfy all our objectives. To become more efficient, the architecture needs to be generalized and materialized using new methodologies. The proposed architecture, as well as the use of QoE in the architecture, are presented in the following chapter.

CHAPTER 4 QUALITY OF EXPERIENCE-BASED PROVISIONING SERVICES IN VEHICULAR CLOUDS

4.1 INTRODUCTION

Vehicular Cloud Computing (VCC) can transmit relatively large volumes of communication data, and provide on-demand applications and services based on the pay-as-you-go principle. The concept is aimed to be widely adopted, which is highlighting many research issues and challenges. From a driver's point of view, privacy, service cost and service latency are the most critical challenges.

Providing vehicle drivers with provisioning services at low cost, with as little information as possible revealed to service providers and minimal service latency, can be difficult due to the peculiarities of vehicular cloud computing. A detailed discussion of such peculiarities can be found in [9][76]. As discussed in previous chapters, researchers are attempting to address these issues independently, without considering the effects they have on one another. There are important interactive relationships between service latency, cost and user satisfaction in a cloud environment, and the impact of each on the other should not be overlooked. Cost can be the most important factor for drivers when they are deciding whether to use the vehicular provisioning services. Moreover, service providers have full control of the data in a cloud, and users are unaware of the physical location where the services are executed. Furthermore, the fact that they are not aware of how the information they provide is processed, stored or secured in the cloud. Therefore, drivers are reluctant to disclose their personal information to unknown providers. Ensuring the privacy of driver's data when dealing with or switching between service providers is imperative, as users must be confident that their information will not be used inappropriately, or sold to a third party. Thus, the privacy of vehicular cloud user data (either raw or processed/mined) is a significant concern.

If vehicle drivers are provided with a rich interactive environment, one that allows them to have input on decisions regarding the quality and level of satisfaction of the provided

services, their feedback will improve the provisioning quality and promote the technology itself. The objective of this chapter is to provide a Quality of Experience (QoE) framework that addresses all these concerns together. The proposed QoE with vehicular cloud can be defined as vehicular experience for service provisioning, with the aim of providing the best rating for the best services, assuming that the QoS has been found satisfactory by drivers. QoE is an emerging metric that has been applied to different systems and technologies, and we anticipate that the interactions between a vehicular cloud and its subscribers (i.e. drivers) will be optimized once QoE is adopted. Feedback regarding user experience can be highly beneficial to service providers, current and future users, as well as other systems. Service providers could anticipate a user's interest based on their experience, and thereby enhance the quality of the services. We advocate that our objectives can be met once QoE is applied and adopted within vehicular provisioning services.

The rest of this chapter is organized as follows: We first review the literature to learn how QoE has been used with other systems, and determine if it is practical to be used by vehicular provisioning services. Next, we develop a proposed QoE framework for provisioning services in a vehicular cloud, and extend the VTTP architecture presented in 3.3.2 to incorporate the QoE concept. The QoE-based system procedure is then described in detail, clarified with explanatory sequence diagrams. Finally, we conclude the chapter with the performance evaluation and analysis of the proposed system.

4.2 RELATED WORK

Currently, related work on QoE-based vehicular provisioning services is very minimal, and, to the best of our knowledge, the studies that have been done during the course of this Ph.D. thesis study, represent the first effort to apply QoE-based vehicular cloud service provisioning concept. QoE is still a fairly new approach in telecommunications networks, and it is emergent due to recent technologies like vehicular cloud computing. It has been widely adopted and used in other research and technologies, including multimedia streaming, communication ecosystems, haptic applications and virtual environments and online gaming [83][84][85][86][87]. In this section, we overview the

use of QoE with these systems and frameworks, and highlight the benefits and added value of using it. These experiences will help us introduce the concept to our proposed methodologies for vehicular clouds.

4.2.1 QoE IN COMMUNICATION ECOSYSTEMS

QoE has become a major research theme in communication ecosystems. A communication ecosystem can generally be defined as a multi-disciplinary system model that incorporates technical issues with business models and human behaviour. The broad variety and diversity in technologies, information, and science encourages different entities and technologies to discuss each other's requirements and come together to enhance their interactions. QoE found the bonds among all of it. Applying a communication ecosystem will help meet user requirements and enhance the degree of QoE.

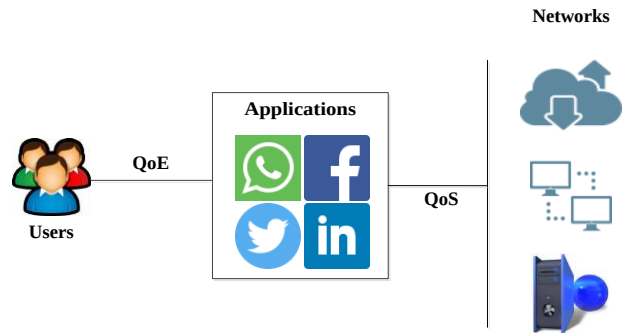


Figure 4.1: Relationship of QoE and QoS in network model

The work presented in [84] reviewed QoE in communications ecosystems, and defined the relationship between QoS and QoE. Its originality and detail will be very helpful for developing future uses of QoE. Figure 4.1 illustrates the relationship of QoE and QoS in a network model, and clearly shows that QoE is related to the users and applications, while QoS manages the network setup and parameters between the applications and the networks. Users' concerns are simple: application quality, delivery availability, and so on. QoE determines the degree of user satisfaction of the provided service while QoS cares with communication parameters like bandwidth, jitter or packet loss. In addition, QoE is highly connected to the user expectations and objectives than network parameters

values. In the paper, the author outlined the role of QoE in a common framework that includes the entire communications ecosystem.

Several models have recently been developed for communication ecosystems. In [88], the authors found that few models integrated all aspects of a communication ecosystem, and proposed a holistic QoE model that brings all disparate pieces of the communication ecosystem together, to create a total QoE.

4.2.2 QoE IN MULTIMEDIA SYSTEMS

Multimedia streaming is one of the main provisioning services for mobile users. Video quality is an important and difficult challenge, and service providers currently provide playback features using a pause-and-wait technique to enhance the streaming bit rate. A more recent technique is to control video streaming by automatically generating video quality, based on the network bandwidth and condition. However, both these solutions may significantly discomfort the end users. Motivated by this and other factors, researchers and industry realize that if users were more involved, the quality of their experience could be enhanced. The authors in [89] performed a subjective experiment to analyze user streaming activities and link them to network performance and user QoE, and found that service delay is the main contributor to user QoE. The delay is caused by the continuous pausing and playing, as well reductions in the screen size. To address this, another technique known as Dynamic Adaptive Streaming over HTTP (DASH) was proposed. DASH is basically a method to change transmitted video quality automatically. In [90], the authors presented the QoE of HTTP adaptive streaming results on perceived video quality, and found that QoE played a major role and enhanced adaptive streaming by incorporating user information and feedback from different layers. Their study highlighted the QoE influence factors of HTTP video streaming. Similarly, the authors in [91][92] studied the impact of video quality on user QoE in dynamic adaptive streaming. The frequent switching between different video quality levels during video streaming could disturb the user and affect their QoE. A series of subjective video streaming tests to determine the correlation between the user QoE and the frequency, type and temporal location of the video switching events were conducted in [91]. Subsequently, the

Switching Degradation Factor (SDF), a novel parameter to capture the correlation, was presented. Three methods to enhance the quality of experience of wireless DASH users were proposed in [92], and simulations and mathematical analysis showed promising results.

Several other authors have proposed various frameworks to improve users QoE-based multimedia services. Researchers in [93] determined that the first step to enhance and promote the provisioning of mobile streaming services is to improve the QoE of multimedia streaming. The novelty of their work was the development of an adaptive QoE prediction model using machine learning algorithms. Machine learning and Artificial Intelligence (AI) are now widely used due to their ability to create real models based on a user's mind and interests, rather than a program or machine. Similarly, researchers in [94] proposed a system model that would help understand user experiences and their relationship with traditional QoS metrics. This work is also considered very promising, since they used results collected from fields such as psychology, cognitive sciences, sociology and information technology to build a multi-disciplinary framework based on distributed interactive multimedia environments.

4.2.3 QoE IN HAPTIC APPLICATIONS AND VIRTUAL ENVIRONMENTS

Haptics applications and technology have changed how humans interact with machines, and evolved to integrating the sense of touch and the importance of user feedback [95]. This integration encompassed different fields, including medicine and rehabilitation, multimedia systems and video gaming. A. El Saddik et al. were the first to investigate this [85][86][95][96][97][98], and they clearly highlighted the considerable advantages of incorporating the QoE and forcing user feedback in human computer interaction.

In [95], they proposed a taxonomy model for haptic user interface applications and incorporated the quality of experience parameters, in order to measure the overall quality of Haptic User Interface (HUI) application. Later, they advanced their work by proposing a mathematical model for evaluating the QoE in haptic based applications [85]. In the model, the authors quantify the QoE of users when they used haptic-based applications. Five different approaches were evaluated for weight determination of their mathematical

model, and the best results were produced by the approach known as Principal Component Analysis (PCA). The authors also used QoE as an evaluation metric for virtual reality applications [98]. They stated that assessing the QoE of virtual reality applications reflects the overall satisfaction of users, and highlights the benefits of using such applications. A taxonomy using a fuzzy logic interface system to evaluate QoE in multimedia virtual reality applications was proposed, and the results demonstrated that the proposed model reflects the user expectations of the application's quality, and is suitable for QoE evaluation.

4.2.4 QoE IN ONLINE GAMING

Online gaming is one of largest on-demand applications ever created. Games are popular everywhere, and attract interest from different domains and ages. They date back to the traditional downloads to desktop machines and laptops, and now they are widely deployed on smartphones and accessible through web browsers. Mobile gaming also produced tremendous sales volumes in the market share, and the huge interest has attracted industry to invest in the field. There are different types of gaming with various levels of complexity, but, regardless, players always want and expect high quality. Thus, research communities are investigating ways to provide users with the best gaming quality.

The authors in [87] proposed a QoE optimization model for cloud gaming, using Ad Hoc cloudlet assistance. The model studies players' behaviour regarding cooperative sharing and optimizes the QoE of the model. It also tests some heuristic solutions to reduce complexity and accelerate encoding in real-time gaming video streaming. The work presented in [99] explored a different direction for Online Social Gaming (OSG), and proposed a model to evaluate the QoE of OSG. Their aim was to understand the impact of socialization on player QoE in OSG. They achieved this by extracting players' objectives as well as their subjectivity, such as whether playing with a friend enhances the player's QoE and how it affects gaming performance. Moreover, they conducted real-time experiments to determine if a specific action correlates to the player's QoE. Their results show that socialization increases player QoE by 10%.

4.2.5 QoE IN VEHICULAR AD-HOC NETWORKS (VANETs)

As discussed in the previous chapters, vehicular provisioning services have attracted significant attention from both academia and industry, and VANET is considered a main branch of vehicular cloud computing. Thus, the support of high quality services and applications in VANET depends on well-planned VANET design and development. The authors in [100] used the QoS and QoE concepts as measurement bases for vehicular Internet access to applications and services, in order to improve the service quality. QoE was used with this research to improve user experience with the services at the application level.

QoE has also been used to improve sharing services among vehicles in VANET, such as video streaming. A QoE-based routing protocol for video streaming in VANETs was proposed in [101], and simulation results demonstrate that the collected QoE data improves video streaming. In addition, user satisfaction and network utilization are enhanced significantly when QoE and the Optimized Link State Routing (OLSR) protocol are combined. Similarly, the authors in [102] proposed a scheme for video streaming in VANET using the QoE to improve the video quality of experience. Their objective was to increase the network coverage and video download speed. Other researchers have also used QoE for multimedia services delivery in VANET [103][104]. In [103], the authors proposed inserting a QoE-monitor unit in each vehicle to measure end-user satisfaction of the video streaming, and report it to an external module (i.e. QoE-controller). They assumed a video streaming scheduling framework over a cloud-based VANET.

In summary, QoE methodology can be considered the most appropriate solution for vehicular cloud computing provisioning services. Deployment of the technology enables service providers to improve resource utilization by incorporating information and feedback from different vehicle drivers, and thereby deliver improved service quality. QoE can also consider users' device capabilities, network conditions and current server load [90]. The use of QoE with vehicular provisioning services will shorten the distance

of expectations and desired objectives between service providers and vehicle drivers or other parties.

4.3 QUALITY OF EXPERIENCE (QoE) IN VEHICULAR CLOUDS

In the areas of telecommunication and networking, QoE has emerged as a more efficient tool than Quality of Service (QoS). It is worth mentioning that QoS was previously the only measurement factor available to determine the quality of delivered services at the network level. QoS was measured by network parameters such as packet loss and delay which are only the technical aspects of network performance; end user experience and satisfaction was not considered. To build an effective QoE model on measurable QoS parameters, the relationships between QoE and QoS must be analyzed carefully [105]. Several recent studies have stipulated the need for QoE information (as discussed in Section 4.2), but from the end user side [106], and businesses such as Uber (an online based transportation network company) have added QoE to their services [107]. UberX is now available to end users, giving them the opportunity to rate services from 1 to 5 based on their satisfaction, and firm action can be taken if the overall rating goes below a stipulated threshold. Service providers must also adhere to a code of respect and professionalism.

The following sub-sections explain the QoE framework supported by the proposed VTTP model presented in Section 3.3. The framework will be used by vehicle drivers to request services in a vehicular cloud. We describe the QoE-based system procedures that are used to formulate QoE parameters, and review QoE-based software architecture and sequence diagrams scenarios.

4.3.1 VEHICULAR PROVISIONING SERVICES QUALITY OF EXPERIENCE (VPSQoE)

In this section, we propose a QoE framework [108][109] to provide several provisioning services in a vehicular cloud at low cost, with the least possible revealed information and minimal service latency. We also extend the framework and incorporate it with our VTTP architecture presented in [77]. The proposed QoE framework, which relies on subjective

criteria rather than objective measurements, considers end users' overall rating of a service, rather than just the procedures used to deliver the service.

The framework is intended to achieve several objectives. The first is to use the QoE concept with vehicular cloud provisioning services to address the challenges of service latency and cost. The second objective is to enhance driver privacy by combining VTTP architecture with the QoE framework to minimize the revealed information, which will reduce risks when drivers switch between service providers. Malicious service providers could easily reuse private information/data without leaving anything tangible that drivers can use to prove the theft happened, so the framework must ensure complete privacy of personal information and stored data. Thirdly, the framework will not integrate feedback received only from vehicle drivers, but that from service providers and trusted third parties as well. The fourth objective is to integrate driver requirements and easily associate them with their requested services. And fifth, to satisfy drivers requirement based on their interest in the services and TTPs.

An earlier version of VTTP architecture intended to reduce service latency that used trusted third parties to help meet some of the driver requirements (e.g. service latency) was proposed, but it did not consider pricing and privacy. In the following, we will elaborate on how driver requirements can be formulated and adapted within our proposed framework, allowing parties to accept online payment services.

Figure 4.2 presents the overall view of QoE framework. The framework uses a Trusted Third Party (TTP) between the drivers (e.g. $u_{1...n} \in U$) and the Service Provider (SP). The QoE is established at the three main communication layers: (1) $QoE_{U \rightarrow TTP}$, (2) $QoE_{TTP \rightarrow SP}$ and (3) $QoE_{SP \rightarrow TTP}$. The framework clearly shows that no experience is exchanged between service providers and drivers. TTPs will be interacting with SPs for all drivers' matters, as highlighted in our earlier proposed VTTP. The TTP layer in Figure 4.2 represents the main core in the proposed framework, and it handles all communications with the service provider and potential users. TTPs are well-known, successful commercial organizations that provide and sell services to users. Users choose a TTP based on two factors: (1) their interest in the requested services, since TTPs specialize in

certain type of services, and (2) the reputation of the TTP for providing such services among peers in the network. Each TTP is assigned a QoE-based reputation value, calculated from ratings received by all drivers that received their services. Drivers have access to these overall ratings, which will help them understand what to expect from different TTPs that specialize in diverse services that drivers need, such as entertainment, professional, medical and so on. Interoperability and standard interfaces between third parties are considered in this framework, with TTPs that provide the same service types grouped and presented to the users as one category. TTPs also have lines of communication between each other to guarantee basic quality of service, and those in the same service type have access to other TTPs that provide different service types, as shown in Figure 4.2. In this framework, QoE is also deployed at the cloud level between service providers and TTPs, and gives SPs and TTPs the opportunity to rate each other. Both parties overall ratings will be visible throughout the system to maintain creditability.

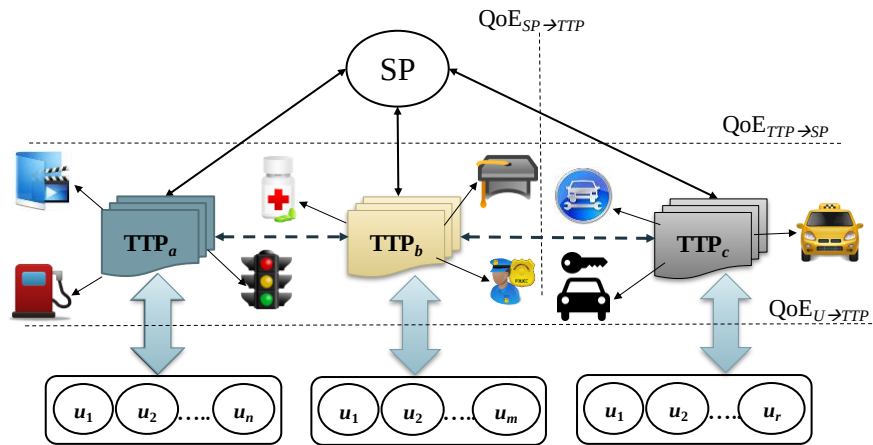


Figure 4.2: QoE framework for provisioning services in vehicular cloud

The framework is also de-centralized and in a hierarchical mode, which provides scalability and minimizes computation overhead. As shown in Figure 4.2, the relationship between users and TTPs is cluster-based, with the TTP as the cluster-head and users in the cluster connected to it. The relationship between the TTPs and service providers is also a cluster-based, with the service provider as the cluster-head and the different TTPs as participants.

The QoE framework must address user requirements based on their interest in the requested services. To do this, we introduced a quality of experience system metric, which is collected and partially formulated from previous users of the vehicular cloud. This metric is a weighted combination of the latency, cost and privacy factors, and is presented and discussed in the following sub-section.

4.3.2 VPSQoE FORMULATION AND RATING

QoE in the proposed framework is formulated as a weighted function of provisioning service latency, cost and privacy (i.e. the amount of information revealed to the service provider), as shown in Eq. (4.1). In the equation, D, P and I represent the latency, cost and revealed information to the provider, respectively. As shown in Eq. (4.2), the sum of the coefficients in the equation is one, and the coefficients represent the driver priority and interest in the corresponding metrics.

$$\text{QoE} = \alpha \cdot D + \beta \cdot P + \gamma \cdot I \quad (4.1)$$

$$\alpha + \beta + \gamma = 1 \quad (4.2)$$

During communication between drivers and TTPs in the proposed system, drivers have the option to prioritize their service requirements. For example, they can choose to subscribe to the TTP that charges less than others, by ignoring the parameters of α and γ in the QoE equation (4.1). The same option can also be applied to latency or revealed information. Thus, all drivers can adjust the coefficients of the QoE function based on their interest. However, if the driver was able to select the best TTP by prioritizing their preferences on all three factors (i.e. cost, latency and revealed information to service providers) it would benefit the entire system. This is what we intend to achieve with QoE, and to do so we need current and previous users' ratings on these metrics in order to determine the overall reputation of the QoE function for each service by any entity in the framework. It is worthwhile noting that the relationship of the QoE equation in this thesis is simply linear due to the fact that we are handling the perspective of user's experience by using numeric rating methodology and do not handling QoS aspects as discussed in Section 4.2.1.

Interactions between TTPs and SPs are critical, since our TTP target is to negotiate with the service providers on service cost, latency and, most importantly, the amount of revealed information. A TTP can use the QoE (i.e., SP reputation) to decide which SP is would be best for a single or multiple service. One of the key factors that the TTP will give drivers to ensure their satisfaction and reflect it in their rating, is the responsibility to provide them with the best cost, least latency and minimum personal information revealed. Privacy will be maximized with this model, since service providers will not have user information or identity, only the personal information that the driver has agreed to share through the TTP. Indeed, it is more beneficial for a driver to deal with a credible TTP than a random service provider, as they will have the option if switching to another service provider in the future. Furthermore, drivers will have more control over their information with one TTP rather than using a different service provider each time they request a service.

The framework also deploys participants' experience at each stage. At first, drivers decide which category of TTP they are interested in (TTP_a , TTP_b , TTP_c), and then choose a specific TTP from that category (e.g. $TTP_{01} \in TTP_b$). Drivers will always have access to an updated list of overall TTP reputation values, which are based on the ratings of previous drivers who requested similar services from those TTPs. Once a driver decides which TTP they want to engage with, a request for the desired service will be sent to that TTP, which means that a new driver has joined that TTP cluster. As soon as the TTP receives the request, they will use QoE about the best SPs in the cloud that provide the service, negotiate with them if required, and provide the best SP fit to the driver, who will then receive the service. Upon providing the service to the driver, the TTP starts charging the driver based on usage, and when the service is complete the driver is required to rate the service with respect to the three QoE concerns: latency, cost, and revealed information. The rating is an integer value in $[1, 10]$ based on user satisfaction where ten and one denote the best and poorest level of quality, respectively. TTPs and SPs are also required to rate each other at a later stage. The rating procedure can be somewhat complicated, so we include sequence diagram scenarios in 0 to make the provisioning services procedure clearer.

System ratings are very important, as they are used by TTPs to analyze and decide which service provider offers the best QoE price/delay/privacy combination, based on user requirements. The ratings will also be helpful when planning longer term engagements with SPs, to ensure they have maintained an acceptable overall QoE reputation most of the time. Similarly, SPs can review the QoE values of TTPs to help them decide if they want to offer their services to that TTP, or reduce or increase the service based on the TTP QoE reputation.

The previous discussion demonstrates how we meet our objectives. Users no longer need to be concerned about their personal information/data when receiving service from a service provider, or even when requesting the same service from different service providers. Because TTPs deal with service providers they are more familiar with them, frequently conduct negotiations, and provide only the minimum driver information required to access the service. In addition, TTPs manage the service charges without sharing additional driver information with service providers, and they are more accountable for their actions than random service providers. Moreover, QoE in this model encourages drivers, TTPs and SPs to maintain their reputations, since all involved parties can access and see them, and they can be a determining factor in whether or not to trust other players.

4.3.3 VPSQoE-BASED SYSTEM PROCEDURE

This section presents the communication procedure between QoE framework entities in detail. Two possible scenarios are conducted (i.e. with and without QoE) to study the impact of QoE on the system model. QoE framework communication procedures start when a driver finds a suitable and available TTP to join, and requests on-demand provisioning service(s). The TTP receives the request and finds the required service, negotiates the terms of the service delivery with the SP on behalf of the driver, and then delivers the service. Charging for the service(s) starts upon the receipt of the service by the driver. Upon completion of the service, drivers are requested to rate of the service cost, latency, and revealed information. All entities in the framework are required to submit their ratings to the system, and the QoE repository of entity reputations becomes

more accurate as more ratings are submitted. The rating is an integer value from 1 to 10 based on driver satisfaction, with ten denoting perfect service and one representing poor service.

Ratings can be used by all parties for various purposes. For example, drivers can access the TTP overall ratings to decide which to engage with, and TTPs can determine the SP that offers the best combination of price, delay and privacy. TTPs endeavour to provide drivers with best QoE combination so they will receive good ratings. The system procedure is also resilient, and addresses a number of requirements and interests. If the user decides to request another service, the coefficients of the QoE (i.e. α, β and γ) function can be adjusted to meet their specific need; such as making cost the priority and not considering the latency or revealed information. Implementing driver feedback on the QoE parameters is a challenging function in the simulation. When a driver selects only one service parameter from the QoE (i.e. cost or latency or revealed information), the feedback is the compilation of the 1 to 10 ratings for that parameter. However, in case of the entire QoE, combinations of the three factors are randomly generated to form the QoE rated value. For example, cost=low=3; privacy=medium=6; latency=high=9. The duration of service usage is considered the price unit. To demonstrate the system procedure in details, the following two scenarios were developed.

A. *System procedure without QoE*

Figure 4.3 presents the sequence diagram of a scenario with no QoE parameters enforced. Drivers in this scenario subscribe to the first available TTP in range, without considering the TTP's experience in finding their desired type of provisioning service. The drivers simply trust the TTP, without taking the experience of previous drivers regarding latency, cost, or revealed information into account.

In a scenario where no QoE is adopted, drivers register with the first available TTP by providing basic personal information, including name (x), credit card information (y) and location (z). The TTP then decides to accept or reject the request; if accepted the driver sends the service request to the TTP and waits for the service (S). The TTP contacts SPs and asks for the requested service, providing some of the client's information to make his

request official. At this level, the TTP is not concerned about relaying personal information to the SP (i.e. x , y and z), since the user has no interest in QoE. Once an SP agrees to provide the service (S) the TTP notifies the driver, who then connects with the SP to receive the service. Once finished receiving the service the driver notifies the TTP, who signals the SP to stop the service. Finally, the SP charges the TTP based on the type and duration of the service. Upon receiving the SP charges the TTP sends them, and the TTP service fees, to the driver.

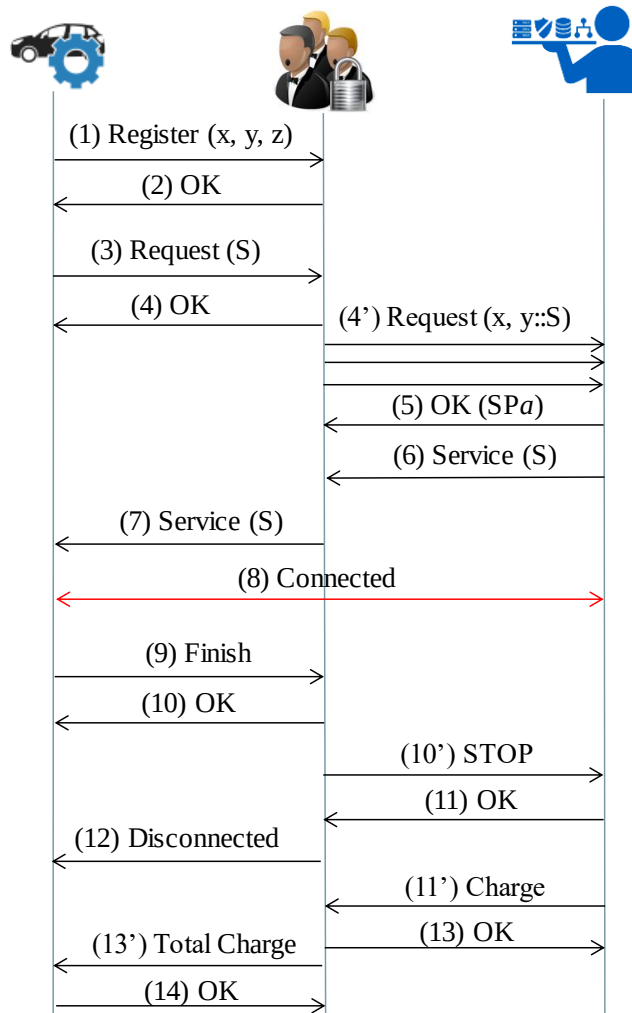


Figure 4.3: Scenario A: System procedure without QoE interest

This basic scenario offers no advantages to the drivers, even though TTPs are supposed to provide better service than SPs alone. Though they can improve the service quality

provided to the driver, TTPs and SPs have no strong motivation to do this. In addition, this approach could actually increase the total fees compared to when no TTP is involved.

B. System procedure with QoE

In contrast to scenario A where no QoE is adopted, Figure 4.4 illustrates the sequence diagram for the QoE system procedure with the following entities: Two drivers, two TTPs, and one SP. One driver is a regular user and the other is a business user, and each considers two TTPs from different categories of professional services. The scenario assumes that service ratings have been collected from previous drivers, and the QoE of each entity has been determined. Thus, each TTP has their own QoE reputation value, which is available to the drivers. The rating is analyzed based on the QoE requirements for a new driver, and the rating database provides the TTP group that best matches the driver's QoE requirements.

With this scenario, drivers can register directly to the TTP that meets their requirements best. This is a clear advantage, since browsing time and random selections are eliminated when adopting the QoE model. Thus, if D_{01} and D_{02} register with their TTPs, and provide them with the required information (i.e. x , y , a , b , S_a and S_b), the TTPs can start the provisioning service process. At this point the TTPs use local SP databases to determine who has best QoE reputation value to provide S_a and S_b , and if the driver does not wish to prioritize the QoE factors the TTPs assume that they are interested in all the QoE aspects. SPs are then contacted and instructed to provide the services to the drivers. Ratings from all parties will be compiled once the users finish using the service. Later, D_{01} registers with a TTP and asks to be charged the minimum by setting up the QoE price factor as shown in Figure 4.4. In addition, drivers can decide how much information they are willing to reveal, which could affect the privacy or latency factors. In the scenario, D_{02} registers with a TTP but provides less information than the first time, and the driver information (x , y , a and b) is not revealed to the SP. And if required, TTPs can choose to provide virtual information of the driver's original personal information to the SP, thereby ensuring that driver identity is not revealed.

If we compare the two scenarios, adopting the QoE system model is the most effective approach, and it also requires less computation overhead. Drivers know which TTP they should register with, and TTPs know which SPs provide the best quality services. QoE will also guide them through selection procedure. In addition, QoE provides different categories of TTPs, many with specialties, and drivers can prioritize their requirements based on their interest. For instance, the drivers who are more concerned with privacy can purchase extra privacy measures and have none of their information revealed.

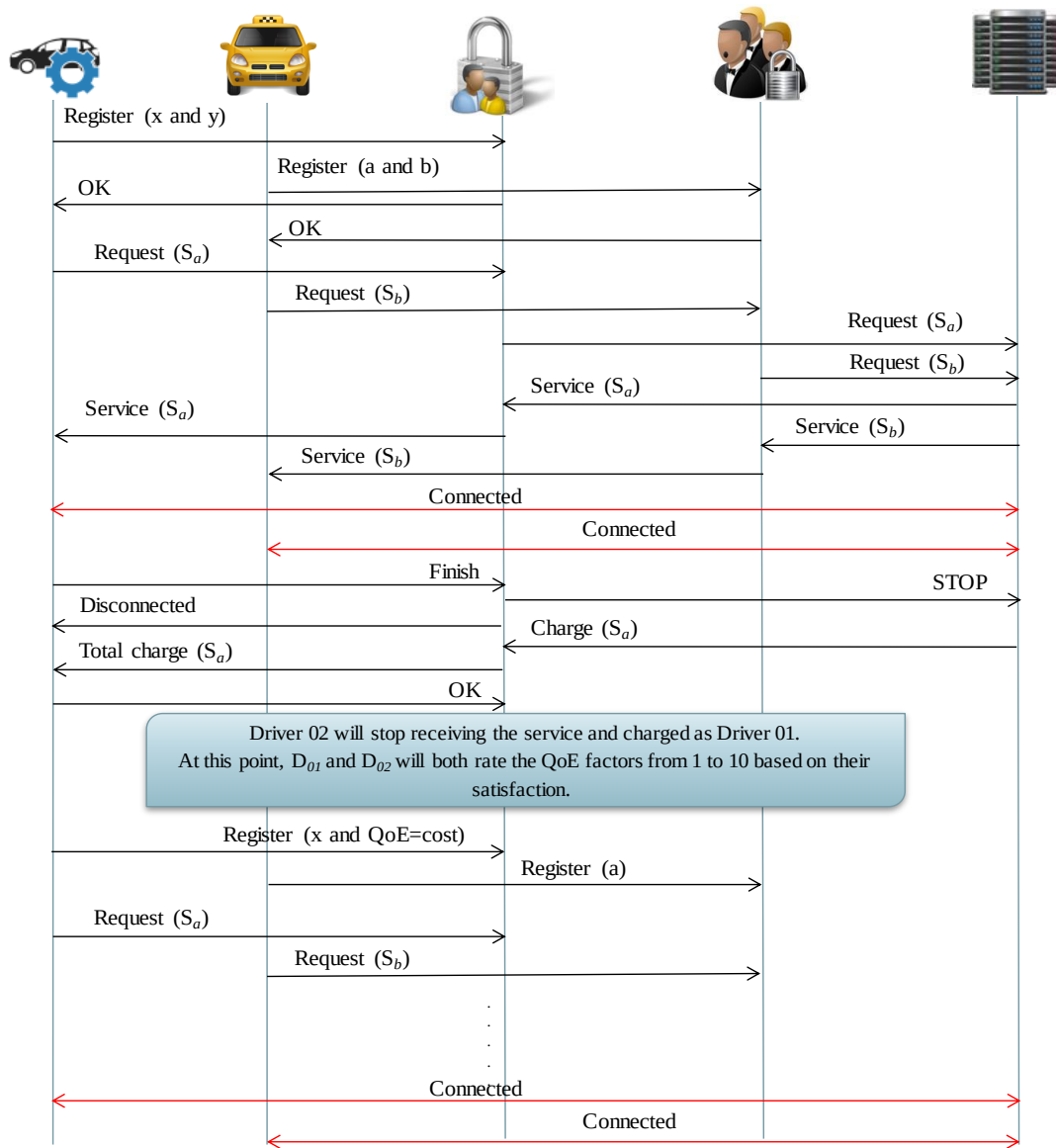


Figure 4.4: Scenario B: System procedure with QoE interest

4.3.4 VPSQoE-BASED SOFTWARE ARCHITECTURE

Figure 4.5 illustrates the QoE-based overall architecture. The architecture has three main software functions, one for each layer in the QoE framework of drivers, TTPs and SPs. Here, we focus only on the TTP functionality, and disregard the SP or end user functions. Vehicular cloud user will be equipped with many on-board resources and tools to simplify their access to provisioning services. Similarly, we do not need to consider the software or platform-specific aspects of cloud service provision, as this is outside the scope of this research.

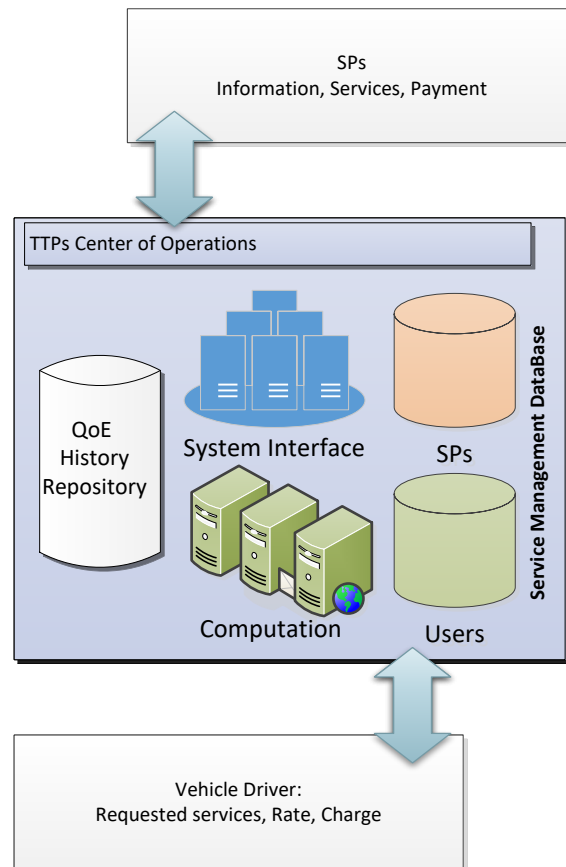


Figure 4.5: VPSQoE-based overall architecture

The lower module of this software architecture is fundamentally related to *vehicle drivers* who are connected to the grid via on-board Internet access. A vehicular web browser with high-speed Internet connectivity is required to guarantee stable provisioning services.

On-board passengers can use mobile applications downloaded to their smart devices to access the network and browse TTPs and their services.

The core of this architecture is the central *TTP* module, which is between the *vehicular driver* and *SP* modules. This entity plays the roles of mediator and contractor for the drivers. Each TTP is responsible for communication with the SPs, and providing the requested services to the drivers. To achieve this goal, we designed the TTPs to use a contact center of operations with five main components: SP database, driver database, infrastructure, system interface and QoE history repository. These components are essential for the design of our QoE framework, and in order to meet our objectives. Two independent databases will store and retrieve drivers and SP information, and a system interface is included to facilitate user interaction with the TTP system. The databases are connected with the TTP infrastructure and system interface. The QoE computation processes for all provisioning services take place in the QoE history repository, where the QoE weighted metrics for each driver and SP are computed and sent after each delivered service. This component is the central control unit of the architecture, since system communications and decoupled computation enable a more generalized design.

4.4 SIMULATION SETTINGS AND RESULTS

To simulate the proposed VPSQoE framework, we used Network Simulator 2 (NS-2) [78], a very popular discrete event simulator running at the networking level. The NS-2 provides results that are very close to real-world expectations and the choice of using NS-2 has been explained in the previous chapters. Our main objective in this evaluation is to examine and test how the VPSQoE framework functions under different approaches, and to compare the results to determine the impact of the QoE on the main driver concerns of service latency, service cost, and privacy. Through simulations, we show that the proposed VPSQoE framework can make a compromise between non-QoE-based approaches and those where driver concerns about provisioning services involve only one of the QoE factors (i.e. latency, cost or privacy). The following sub-sections discuss simulation settings and approaches, mobility models associated with the approaches, and preliminary results and analysis.

4.4.1 SIMULATION SETTINGS

The communication medium of this simulation is similar to that used in VTTP architecture, and we are using the same communication medium to study the effect of incorporating QoE with VTTP under the same simulation parameters. The simulation employs the following parameters: Destination-Sequenced Distance-Vector Routing (DSDV) protocol on an IEEE 802.11.p communication stack with 1024 byte/packets, and a 2Mb/s data rate at a frequency bandwidth of 204 GHz. In the following set of simulations, each was repeated 10 times with a range of 5 to 25 vehicles.

Three different approaches have been evaluated and proposed: *Neutral mode*, *Sensitive modes*, and *QoE-aware*. *Neutral mode* is carried in the VTTP architecture as discussed in 3.4.1, and no QoE aspects are considered with this mode. VTTP model presented in CHAPTER 3 has been used as the benchmark to validate the collected results. Vehicle drivers simply connect to the first available service provider to request service(s). *Sensitive modes* are when the driver is interested in only one factor of the QoE; that is, latency (*Delay-sensitive*) or cost (*Economy-sensitive*) or revealed information (*Privacy-sensitive*). *QoE-aware* mode takes advantage of the three QoE factors, and assumes vehicle drivers are interested in the overall QoE. In order to provide general and precise results with *QoE-aware*, the mode is tested using static and random QoE coefficient factors in each set of experiments. These approaches help to: 1) study the impact of the QoE on service latency, cost, and privacy in different scenarios, 2) study the impact of incorporating the concept of TTP with QoE in vehicular clouds, and 3) to investigate how mobility could affect these approaches.

4.4.2 MOBILITY MODEL

For the sake of simplicity at this point of the research, we used a basic random waypoint movement model, which is commonly used in mobility models in network simulation, to manage driver mobility movement in the proposed framework. The spatial distribution of network nodes in this model is non-uniform [110]. It should be mentioned that there are other realistic mobility models in the literature. However, the aim of this study is to investigate the performance of the VPSQoE framework in vehicular clouds; the impact of

vehicle trajectories is included in the future research agenda. Thus, vehicular node movements are random throughout the simulation, the node destinations are randomly assigned, and each node identifies a destination and moves toward it. Upon reaching the destination, the node stops for a random period of time, known as ‘*pause time*’, then identifies another random destination and repeats the process until the simulation is complete. To demonstrate the effect of this model, we simulated multiple scenarios with a random number of vehicular participants to assess the behaviour of the proposed framework.

4.4.3 SIMULATION RESULTS AND ANALYSIS

The system is evaluated using the three metrics of *delay*, *cost* and *privacy*. Delay is the end-to-end time before a user starts receiving the requested service; cost is the total fee drivers are charged based on their usage, which includes service provider and TTP fees; and privacy represents the total amount of information driver are required to provide in order to receive the requested service. We evaluated these metrics in the following two scenarios:

A) STATIC QoE COEFFICIENTS

Table 4.1: Static values of QoE coefficients

		α	β	γ	SPs	$TTPs$	$Vehicles$
<i>Delay</i>	QoE-aware	0.4	0.3	0.3	4	3	25
<i>Cost</i>		0.3	0.4	0.3			
<i>Privacy</i>		0.3	0.3	0.4			
<i>Delay-sensitive</i>	Sensitive modes	1	0	0	4	3	25
<i>Economy-sensitive</i>		0	1	0			
<i>Privacy-sensitive</i>		0	0	1			

In the first set of simulations, we selected four SP (SP_{01} , SP_{02} , SP_{03} , and SP_{04}), three $TTPs$ (TTP_{01} , TTP_{02} , TTP_{03}), and 25 vehicular users (u_{01} to u_{25}). We initially assumed that vehicular drivers are arbitrarily connected to TTP_{01} , TTP_{02} or TTP_{03} , and that all $TTPs$ are aware of the SPs services. Static coefficient factors values ranged equally between

QoE factors. Table 4.1 shows the static values of the QoE coefficients used in this simulation.

Figure 4.6 depicts the service latency performance of the three different modes described above. As expected, the *Neutral mode* with no interest in QoE had the least provisioning service latency, while the other two modes (*Delay-sensitive* and *QoE-aware*) increased the delay. These increases are expected and can be difficult to avoid. There are many potential causes for increased service latency, such as TTP adding an extra layer of communication between the drivers and providers. Mobility can also increase the service latency by producing increased computation overhead. The average delay difference between the three modes is in the range of 8% to 15%. The *Delay-sensitive* mode should be introducing less average delay than *QoE-aware* mode. The extra delay of *QoE-aware* compared to *Neutral mode* is not optimal, however the behaviour of other two factors was not yet determined.

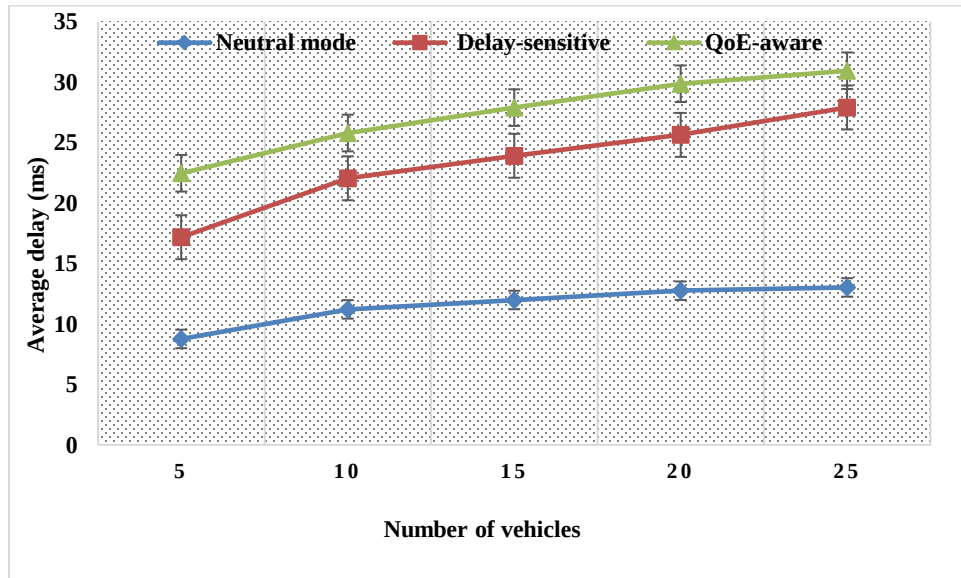


Figure 4.6: Performance of average delays - VPSQoE modes – Static

The second set of experiments assesses the performance of the service cost under three different approaches, as shown in Figure 4.7. Similar to the experiments above, *Economy-sensitive* mode users are interested in the service cost only. And since *Neutral mode* does not consider QoE it has the highest average service cost, while *Economy-*

sensitive mode has the lowest. *Economy-sensitive mode* outperforms other two because β in the QoE equation is set to 1. In terms of average service charges per user, the *QoE-aware mode* is a compromise between the *Neutral* and *Economy-sensitive modes*. Thus, it is possible that drivers could accept some extra delay to decrease charges.

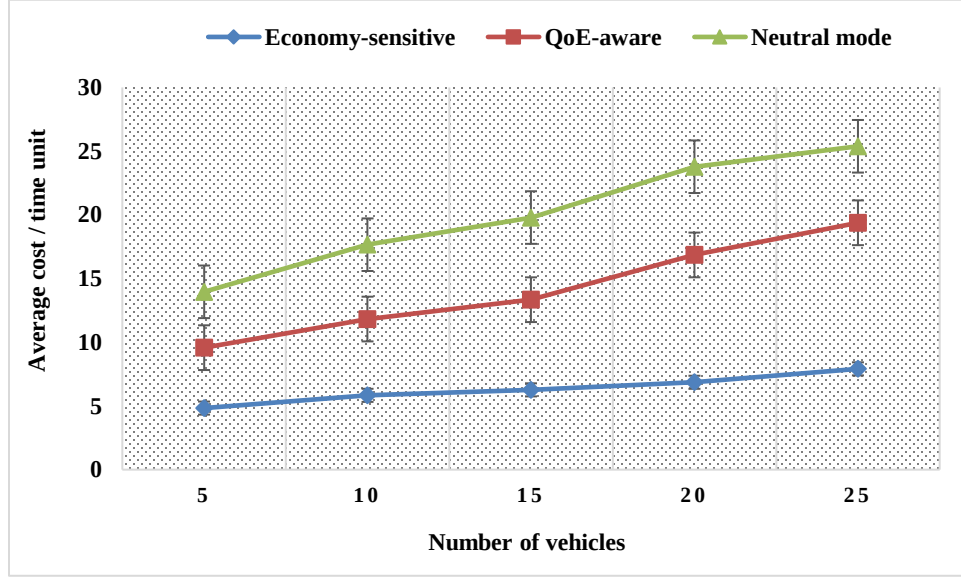


Figure 4.7: Performance of average cost - VPSQoE modes – static

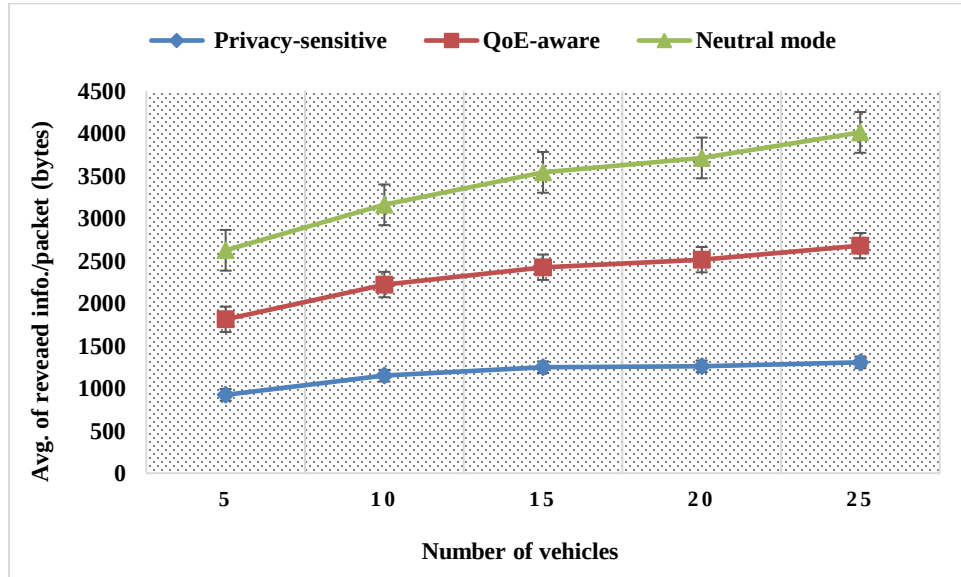


Figure 4.8: Performance of average revealed information - VPSQoE modes – static

The last set of experiments in this sub-section involves vehicular driver's privacy, as shown in Figure 4.8. The only difference between the operation modes here is that

Privacy-sensitive mode drivers are only concerned with the amount of revealed information. The *QoE-aware* approach improved the total amount of information revealed, compared to the *Neutral mode*. The results indicated that vehicular drivers with serious concerns about their personal information/data should adopt the *Privacy-sensitive mode*, since it reveals the least information. The *Neutral mode* showed the poorest distribution of the three modes.

To summarize, using static QoE coefficients means drivers can expect extra service latency, while service cost and privacy are maintained. Sensitive modes outperform all other modes in terms of service cost and privacy, but not with regards to service latency. There could be many reasons for this, since the mobility model, static coefficients and TTPs can all increase latency.

B) RANDOM QoE COEFFICIENTS

In the second set of simulations we randomized everything, and used the same mobility model and VTTP simulation scenarios as earlier. However, this time the scenarios were associated with random QoE coefficient factors. We initially selected ranges of 3 to 22 service providers, 2 to 19 TTPs and 5 to 25 vehicles, and we assumed that the vehicles were arbitrarily connected to the TTPs and the TTPs were aware of all the service providers and had access to their services. Table 4.2 shows the details of the VPSQoE simulation scenarios. The *Sensitive modes* by nature carried the same QoE coefficients whereas *QoE-aware* coefficients are set at random at the beginning of each service request.

Table 4.2: VPSQoE simulation scenarios

<i>Scenarios</i>	<i>Vehicles</i>	<i>TTPs</i>	<i>Service Providers</i>	<i>Sensitive modes coefficients</i>	<i>QoE-aware coefficients</i>
1	5	2, 3	3, 4	Accordingly	$a = 1,$
2	10	3, 5, 6	5, 7, 7		$\beta = 1,$
3	15	5, 9, 9	9, 12, 13		$\gamma = 1,$
4	20	9, 11,	10, 13, 17		$a = random,$
5	25	14, 16,	18, 20, 22		$\beta = random,$
					$\gamma = random.$

We evaluated the proposed approaches using the same metrics of *delay*, *cost* and *privacy*, and we revealed four types of information to the SP: I_a =name, I_b =address, I_c =current location and I_d =credential (e.g. credit card, bank account number). TTPs randomly ask for these types of information, and they typically provide SPs the minimum needed to protect vehicle driver's privacy.

Figure 4.9 shows the performance of the average service delay under the three different modes in the first group of the simulation, and the results yielded two observations: 1) *Delay-sensitive* mode outperforms *Neutral mode* in the five network dense, which was not the case in A), and 2) the *Delay-sensitive* and *Neutral modes* cause the least service delay compared to *QoE-aware* at the early stage of its simulation. However, when the number of vehicles increased to approximately 20, the total average delay was reduced significantly. This is because the service requests increase the feedback and driver experience, which results in improved QoE throughout the entire system. This was a significant improvement when compared to the service delay results obtained in A), and was the best service delay recorded when the network density was 20 vehicles.

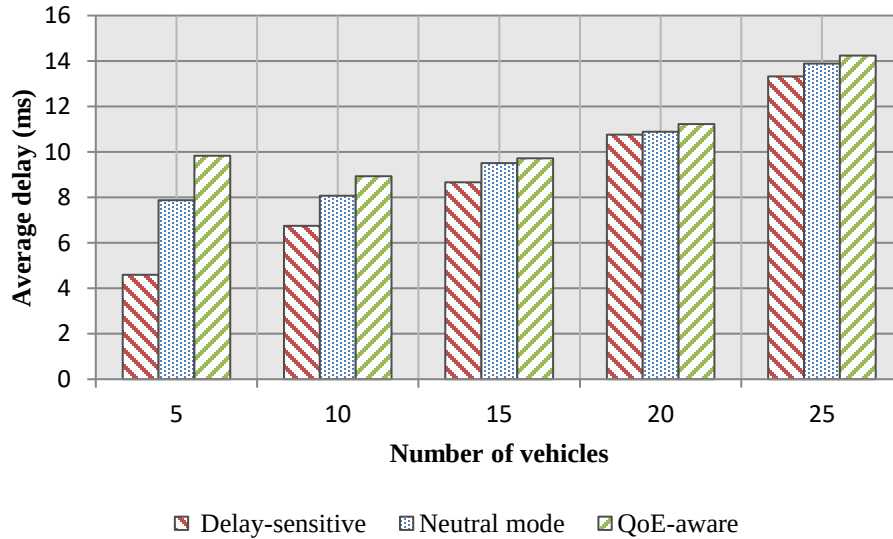


Figure 4.9: Performance of average delays - VPSQoE modes – random

The performance of the average service cost in the second group of the simulation is shown in Figure 4.10. The *Neutral mode* is not considered at this point, since the other

two modes outperformed it significantly. In the *Economy-sensitive* mode vehicle drivers are interested in minimizing their service cost, and in all the system scenarios this mode had the highest service cost savings. Figure 4.10 also shows an enhancement of the *Economy-sensitive* and *QoE-aware* modes when the network density increases, due to the recorded QoE ratings. Increased QoE rating records in the system help with the upcoming selection procedure. Thus, it is more efficient for the vehicle drivers to adopt *QoE-aware* over *Economy-sensitive*, since the charges can be very small.

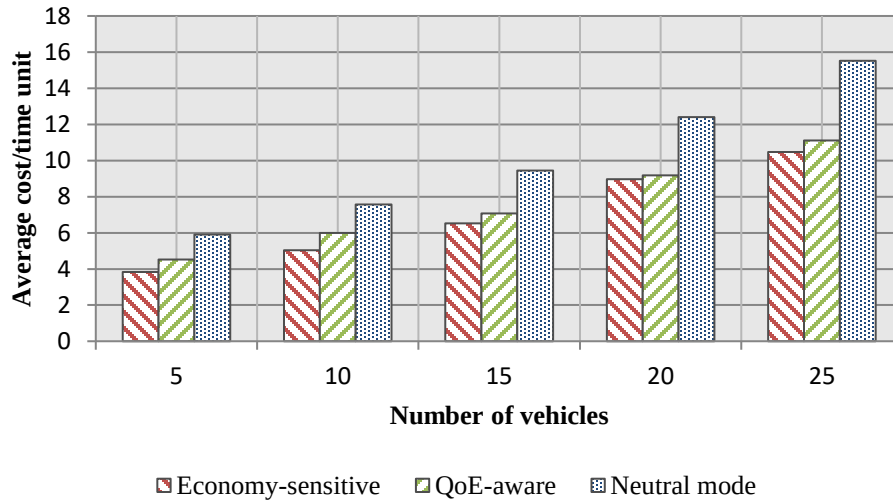


Figure 4.10: Performance of average cost - VPSQoE modes – random

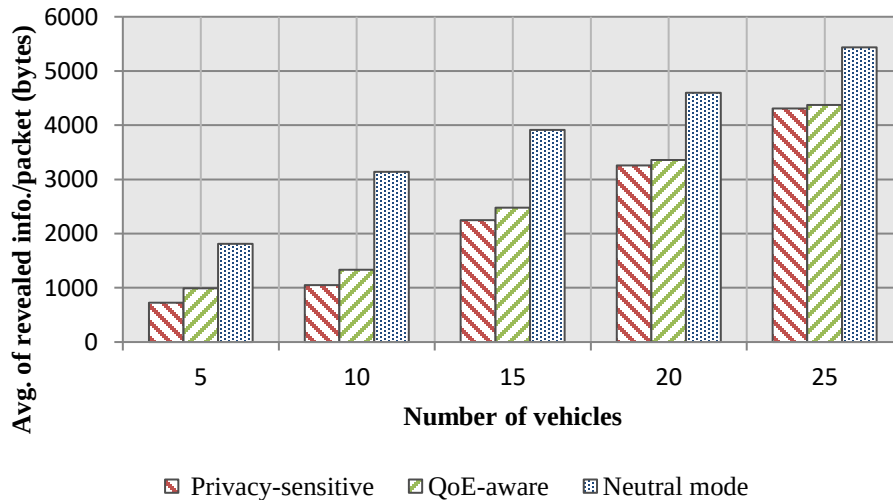


Figure 4.11: Performance of average revealed information - VPSQoE modes – random

In the last group of the simulation, the performance of the average revealed information is shown in Figure 4.11. *Privacy-sensitive* mode represents the case where vehicle drivers are very concerned about their privacy, and uses I_a =name, I_b =address, I_c =current location and I_d =credential (e.g. credit card, bank account number). Figure 4.11 highlights the enhancements due to the amount of the information revealed. *Neutral mode* was not compared with the other two modes, as it revealed the most information. As with the average service cost, *Privacy-sensitive* and *QoE-aware* modes show very similar results at high network density.

4.5 SUMMARY

From a driver's standpoint, service cost, service latency and privacy were identified as the most important challenges to be addressed. In this chapter, we highlighted the significance of QoE in vehicular cloud systems, and reviewed the most relevant related work. We proposed a QoE framework for vehicular provisioning services that utilizes the concept of TTP between vehicle drivers and service providers. The objectives of the framework are to provide provisioning services in vehicular clouds at minimal cost, latency and revealed information. Three different approaches were proposed and evaluated under this framework: *Neutral mode*, *Sensitive modes* (*Privacy-sensitive*, *Economy-sensitive*, and *Delay-sensitive*), and *the QoE-aware mode*. We formulated the QoE based on vehicle drivers' levels of satisfaction, which were collected with a rating system. The VPSQoE framework has the flexibility to switch between TTPs, and while switching drivers can exploit the benefits of the QoE ratings and connect to the TTPs that best meet their specific requirements. We evaluated the proposed framework via simulations, and a random waypoint mobility model was associated with the framework. Numerical results showed that QoE-based provisioning services in a vehicular cloud can be a compromise between basic provisioning services and other approaches meant to meet a specific driver requirement (i.e. cost, latency or revealed information). The QoE framework still needs work, including investigating more effective mobility models, and optimizing the TTP and SP selection procedures. These issues are addressed in the following chapter.

CHAPTER 5 AN AUCTION-DRIVEN MULTI-OBJECTIVE FRAMEWORK FOR PROVISIONING SERVICES IN VEHICULAR CLOUDS

5.1 INTRODUCTION

The advent of cloud computing and the benefits presented by smart mobile devices are leading to the wide adoption of the mobile cloud computing paradigm. In a cloud system, various computing resources (e.g., computation, storage, networking, etc.) are consolidated on virtual infrastructures and offered to the end users based on the pay-as-you-go fashion [111].

Mobile Cloud Computing (MCC) consolidates cloud computing and mobile networking, and defines a new business model between mobile users, mobile network operators and cloud providers. Storage and processing tasks are placed on a cloud platform which is accessible to smart mobile devices. By doing so, computing power and data storage overhead costs are eliminated at mobile devices [10]. All users' applications are then accessed wirelessly through web browsers on mobile devices. Vehicular Cloud Computing (VCC) denotes a shared pool of computing resources in a set of wirelessly connected vehicles, which are available for on-demand and rapid access based on the pay-as-you-go fashion. VCC consolidates the benefits of MCC and vehicular communications.

Vehicular cloud provides a number of on-demand resources to vehicle drivers while en route. The deployment of such resources is very promising in terms of the wide scope of applications and provisioning services that can be provided. Nowadays, providing provisioning services to vehicle drivers is rather important due to several reasons. To improve driving experience, make vehicles more informative, reduce accidents and save lives, make access to services and information faster and easier, are just to name a few. However, providing provisioning services to vehicle drivers are still facing many challenges [108][109].

Prior to the VCC concept being widely adopted, several research issues and challenges need to be carefully addressed in such environment. Drivers' privacy, service cost and service latency are identified as the most crucial challenges to be addressed. State-of-the-art includes several proposals to address these issues however a combined solution that aims to address these requirements holistically does not exist yet. The main issue with the privacy and security is the lack of control over the data which is stored and/or processed on virtual resources [112]. In a vehicular cloud system, users prefer to have more control on the decision of what information could be exposed to third parties (e.g. service providers) and what information should be kept private. Besides security and privacy, service latency and service cost are further concerns for the drivers [13][15][113][114]. That being said, these parameters affect each other in on-demand provisioning service. Thus, we sustain that such challenges must be studied together, independent solutions to those issues cannot meet and fulfil driver's satisfaction, in most cases, drivers are more concerned with more than one of these factors. So far, very little attention has been devoted to tackle these issues all together.

In the previous chapters, we identified the impact of utilizing a trusted third party between vehicle drivers and service providers and a VTTP architecture has been proposed. Furthermore, a Quality of Experience (QoE) framework based provisioning service in vehicular cloud (VPSQoE) has been developed. The proposed framework shows a promising result in reducing service latency, preserve driver's privacy, and enhance service cost saving. To the best of our knowledge, VPSQoE is the first developed framework that considers vehicle driver's quality of experience to promote vehicular provisioning services. We are anticipating that this work still can be improved and better results can be achieved if the selection procedure of the system participants (drivers, trusted third parties, and service providers) has been handled automatically which will maximize participant's revenue and better address their interest requirements.

Our main objective in this chapter is to use the concept of the auction theory to provide provisioning services to vehicle drivers at low cost, with minimum amount of the revealed information, and low service latency. We hypothesize that the use of the auction theory permits the Vehicle Drivers (VDs) to benefit from the different available services

with the same applied cost. TTPs can also maximize their profit if they been able to select the group of participants they can serve them better. Our auction participants are mainly the VDs, TTP, and Service Providers (SPs). To this end, in this chapter, we present two methods to compare the obtained results: *First, Naïve QoE-based Provisioning (NQoEP)* approach which collects the requirements of newly joining drivers, as well as the experience of the drivers who have already been served by the vehicular cloud. Through these values, we define Quality of Experience (QoE) value for each Service Provider/Trusted Third Party (SP/TTP) tuple. A negotiation server matches the drivers request with the SP/TTP tuple that would offer the best service based on the driver's requirements and the QoE values assigned to each tuple. It is worthwhile noting that the QoE in this chapter denotes participant's recommendations and rating based on their past experiences with latency, service cost and privacy. An essential assumption is that the users report their experiences with the TTP/SP tuples truthfully. NQoEP approach builds on our previous works presented in CHAPTER 3 and CHAPTER 4. *Secondly*, we propose an *Auction-driven Quality of Experience-based Provisioning (AQoEP)* approach by extending *NQoEP* to incorporate multiple SPs and TTPs, and include the concept of auction theory to select, negotiate, and serve on behalf of the participants. We show that the *Auction-driven Quality of Experience-based Provisioning (AQoEP)* enhances the overall system performance by finding the best vehicle-SP-TTP matching.

The reminder of this chapter is organized as follows: The related work on provisioning services in vehicular cloud is presented in Section 5.2. Section 5.3 overviews the QoE framework of vehicular provisioning services. The proposed auction scheme is presented in Section 5.4 including system model along with the software architecture. Mobility model, simulation settings and numerical results are presented and discussed in Section 5.5. Finally, we conclude the chapter in Section 5.6.

5.2 RELATED WORK

Vehicular clouds offer a wide range of benefits for various environments and applications, though many open issues and challenges remain unresolved. In this section, we review the main challenges and notable solutions.

The authors in [19] classified the issues that emerged as a result of employing security and privacy in a vehicular cloud. In order to produce a system model for all vehicular technologies, integration of security/privacy features should be a part of the communication stack of any system platform.

Security and privacy issues in vehicular communications have been explored by many researchers in academia and industry. Various solutions for these concerns have been proposed, including pseudonym identity, anomaly detection schemes, public or anonymous keys and digital signature verification, and have been widely investigated [20][43][115][116][117]. In [43], the authors proposed a framework called personal data vaults, which was designed to control and protect the stream of users' personal data. The use of this framework allows only the main owners access their data, and though it is an individually controlled method for data repositories it does not guarantee users' anonymity. One of the main objectives of privacy is to protect user identity, and sometimes hide it, yet there is no viable solution to address the anonymity issue in vehicular clouds. Researchers in [115] studied the importance of information collection in smart cities, and identified the privacy threats. They proposed a privacy-enhancing architecture using an adaptive pseudonymization technique, to provide real-time awareness and enhance privacy security.

Pseudonym identity [20][116][117] has been considered as a solution for protecting user identities. In [116], the authors proposed a protocol based on pseudonymity to reduce the possibility of discovering the identity of drivers from their sent data. Such a protocol could be considered for communication between vehicles when they share resources and data en-route. A feasible protocol that enables resource and data sharing between vehicle drivers and cloud providers has not yet been developed. Similarly, the authors in [20] proposed a non-pseudonym strategy based on Tamper Resistant Hardware (TRH) to avoid proliferation of vehicle identities. This approach has a negative impact on routing efficiency, and handling and discovering malicious vehicles trying to get the benefits of these protocols has not been addressed. Anomaly detection schemes can be considered for such problems [118][119]. Anomaly detection schemes can be used for data analysis, and to identify suspicious sources, monitor the normal behaviour of the network flow,

and protect the vehicle network from potential attacks. The authors in [118] presented a detailed study of several anomaly detection schemes that could identify possible network intrusions. Anomaly detection can also be deployed to monitor vehicle network security but continuous monitoring of network flow to identify suspicious sources could have a negative impact on latency, and lead to increased network overhead.

Other studies assessed the benefits of using public key encryption with keyword search (PEKs), and searchable encryption public key techniques. Key certification in PEKs is a complex process [120] as the public key certification is frequently updated, which could lead to communications overhead. Such a framework is also potentially vulnerable to inside keyword guessing attacks (KGA). Searchable encryption public keys [121] propose using a dual-server PEKs framework to address the vulnerability. It is important to note that encryption/decryption processes should be considered carefully in order to prevent computation overhead or potential delays.

Transferring data from service providers to vehicles without excessive delay is an issue that needs special attention, because a reasonable cost for vehicular service requests and data must be ensured. Cost efficiency and bounded delay are among the major challenges impacting the adoption of vehicular clouds. Vehicular clouds are also a major concern with respect to the mobility issue. Unpredicted moving vehicles produce countless mobility scenarios which potentially increase service delay and possibly service cost. With poor Internet connectivity, these scenarios are likely, and require special attention in such an environment.

Several researchers worked on cloud service pricing [21][122][70][74]. Some of these proposals should be revisited and tailored to vehicular clouds. To the best of our knowledge, the issue of cost efficiency in a vehicular cloud environment has not been investigated yet. In [21], the authors proposed efficient dynamic scheduling to enable energy savings and reduce delay. A pricing mechanism to optimize mobile users and service providers and reduce their total costs for both non-cooperative and cooperative scenarios was suggested in [122]. In [122], a message dissemination scheme for VANETs was proposed to provide high message delivery ratio and decreased delays. A directional

greedy approach creates a group of candidate nodes that hold the message to ensure optimal reliability. The authors in [123] proposed a repetition-based broadcast protocol for reliable broadcasting that guarantees a minimum number of broadcasts by signalling neighbouring nodes to transmit the same message at the same time. This scheme relies on cooperative diversity and a virtual antenna array. Similarly, in [124] the authors proposed a contention-based packet forwarding scheme for data dissemination in VANETs that introduces lower network overhead. This, in turn, helps decrease delays.

The studies mentioned above prove that there are no collaborative works in process that jointly address all these challenges. Exploring the shortcomings of privacy and security is an integral aspect of this research. Provision service delay and service cost issues in vehicular clouds have been examined less comprehensively, even though they are as important as security and privacy challenges. Although drivers could have serious concerns about service pricing and latency, the state of the art does not offer any legitimate and available solutions for service-price matching. We are expecting that the adaptation of the auction theory concept into the proposed QoE framework is likely to utilize the overall system performance and enhance system selection procedure. In this chapter, we are extending QoE framework supported by auction theory concept in order to meet most driver requirements, and addresses the identified challenges.

5.3 QUALITY OF EXPERIENCE (QoE) FRAMEWORK

Currently, QoE has been materialized and used with different research spaces, each of which introduces its own definition. For example, the International Telecommunication Union (ITU) defined QoE as, “*the overall acceptability of an application or service, as perceived subjectively by the end-user*”[125]. It is also worth mentioning that QoE emerged as a broader concept than QoS. The authors in [126] defined QoE as a multi-disciplinary field available to practitioners to evaluate systems, services or applications independently, or during the design phase. For networking communities, there are several boundaries between QoS and QoE that are not clearly defined; the differences and commonalities between the two can be found in. This section explains the proposed QoE framework to be used with the auction approach discussed in Section 5.4.

Figure 5.1 presents the QoE framework under study [127]. The framework is comprised of three main participants: vehicle drivers, trusted third parties and service providers. The architecture employs a trusted third party between the drivers and the service providers to act on their behalf. The QoE framework is set up in a hierarchical mode, to provide scalability and reduce congestion overhead among the participants. Our auction scheme, introduced in the next section, shows that trusted third parties are cluster-heads, while vehicle drivers and service providers are cluster-participants. For example, a group of drivers and service providers are connected to their cluster-head (i.e. TTP). The TTP receives a service request from a driver, processes it, and delivers it to the particular driver. Cluster participants can choose to join or leave at their convenience. This model is based on our former work in [108][109][127], in which we applied clustering but the implementation was more complicated. Thus, instead of dealing with two cluster-heads at different stages, we simply select one cluster head to handle all the communications at once.

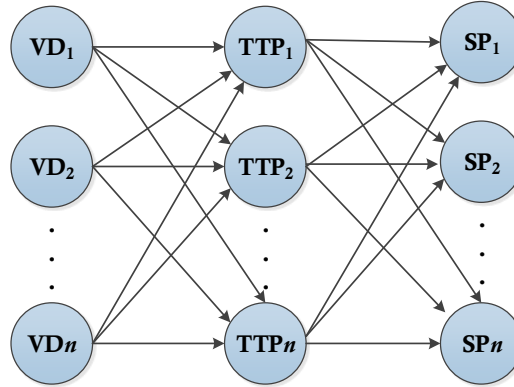


Figure 5.1: The QoE framework under study [127]

The trusted third parties in our architecture are assumed to be well-known commercial organizations that purchase services from service providers and sell them to vehicle drivers. The third party responsible for delivering QoS to its users is trusted and reliable. The responsibilities of a trusted third party include relevant communication aspects, finding the best-fit service request, negotiations the services, searching for the best price, filtering bad or impractical service requests, controlling misbehaviour by participants, and guaranteeing payment.

QoE is a weighted function of service provision delay (i.e. latency), service price (i.e. cost) and information revealed (i.e. privacy), as defined in Eq. (5.1) [127]. In the equation, D , P and I represent the delay, price and information revealed, respectively. As shown, the sum of the coefficients in the equation equals one. QoE components are obtained via feedback from previously provisioned vehicles in a vehicular cloud, and formulated according to a weighted combination of the three key factors.

$$\text{QoE} = \alpha.D + \beta.P + \gamma.I \mid \alpha + \beta + \gamma = 1 \quad (5.1)$$

A driver who requests on-demand provisioning services has the option of joining a trusted third party. The drivers can do this randomly or manually by assessing the available trusted third parties options. It is worth mentioning that each participant in this framework has a QoE weight for every offered service, and the weight can be considered as or refer to the participant's QoS. Typically, a driver requests a service from a trusted third party that will negotiate the service on his/her behalf prior to delivering it. Upon provision of the service to the driver, the trusted third party charges the vehicle driver based on the amount of usage. Thus, service providers receive their payment indirectly from the trusted third party, not from the driver. In this way, service providers will not be able to acquire a driver's identity or any other personal information. More importantly, when a QoE system completes a driver request, each participant provides recommendations based on their experience regarding the three QoE factors of delay, price and information revealed. In [1, 10] the recommendation is a numeric value based on user satisfaction, with ten representing the highest rank. Each party in the QoE framework has his own QoE reputation, which is the total average of recommendations received from other parties for each service provided by/to any party in the system. The recommendations are used to analyze and determine which service provider can offer the best price-delay-privacy combination, based on the driver's preference and requirements. The drivers can adjust the coefficients of the QoE function (i.e. α , β , and γ), or simply select one or more interests over others in the QoE.

Clearly, the QoE framework secures a driver's personal information, but this is not a concern since the trusted third party deals with the service provider. Trusted third parties

provide as little driver information as possible to service providers, and in some cases may not need to provide any information. Moreover, drivers can request services based on their preferences. For example, they could choose fair price but less delay, or lowest price with acceptable delay. Thus, the objectives of the service cost and latency will be satisfied by the QoE framework.

5.4 AUCTION-DRIVEN QUALITY OF EXPERIENCE APPROACH

The idea of using a trusted third party in vehicular cloud (VTTP) has been briefly presented in CHAPTER 3. This architecture potentially meets some of the drivers' requirements in terms of service latency. In a later work (CHAPTER 4), we have extended VTTP architecture to meet all provisioning service requirements of the drivers based on QoE approach. QoE components are obtained via previously provisioned vehicles' feedback in the vehicular cloud, and it is formulated according to a weighted combination of the three key factors: privacy, cost and latency. Communication and engagement procedure between drivers and TTPs explained in the previous chapters does not fully represent real-world. Drivers can be hesitating to which third party they might engage with, thus a dynamic interactive environment in order to facilitate their provisioning services selection can be an asset. An automotive selection mechanisms based on strong approach such auction theory can better accomplish this and reduce possible computation overhead. In this section, we propose an auction-driven QoE-aware provisioning mechanism to assess in finding the best vehicle-SP-TTP tuple.

The auction architecture consists of three potential participants; (1) Vehicle Drivers (VD_1 - VD_n); (2) TTPs (TTP_1 - TTP_n); and (3) Service Providers (SP_1 - SP_n), as shown in Figure 5.1. Those are the same participants of VPSQoE framework. However, this time, QoE is used as the auction key metric in finding the best tuple. We include several TTPs and SPs with several drivers to study the effect of auction complexity. Furthermore, for the sake of scalability, the mobile network between VDs, TTPs and SPs have a hierarchical architecture. For instance, a TTP can be consider the cluster-head and has always a group of connected vehicles and SPs in his auction. Once the service is delivered to a particular vehicle and the driver is charged, auction service rate is recorded,

and the vehicle leaves the TTP auction while another vehicle joins. The following subsections will help us understanding the auction architecture and selection procedure.

5.4.1 AUCTION-DRIVEN QoE-BASED SOFTWARE ARCHITECTURE

The auction software architecture is an integral part of the proposed scheme to understand how different auction components have to communicate among and across software layers. The auction-driven QoE-based software architecture shown in Figure 5.2, uses QoE as its key metric to satisfy driver most wanted requirements. It has three main layers as follows: 1) Vehicular drivers' layer, 2) TTPs layer, and 3) SPs layer.

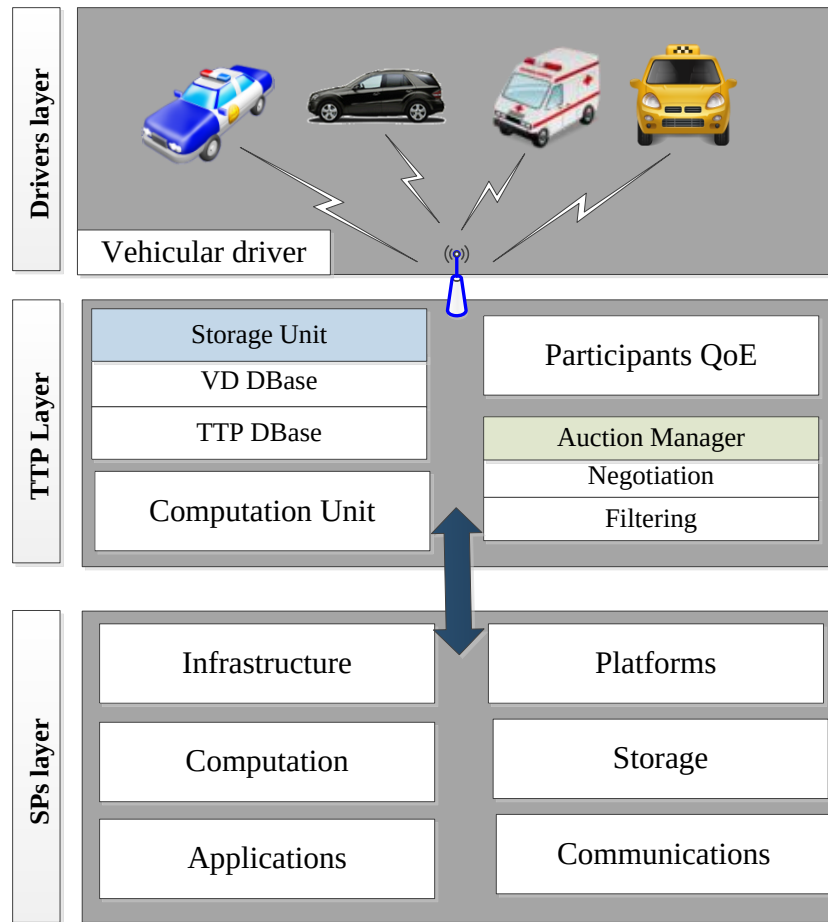


Figure 5.2: Auction-driven QoE-based software architecture

The top layer is basically related to VDs who are connected to the grid. It is expected that the driver is connected to the Internet through a browser. Drivers may also have a mobile application that enables them to access the network and browse through all available

TTPs and their available services and then make manual selection based on their preferences. Such application saves drivers time and provides them with strong tool to choose the best suitable TTPs with best price or best other services such as, privacy or latency. A more suitable approach if drivers are supported with an auction application to run this process on behalf of them. Basically, drivers join a pool of potential auction participants and re-directed automatically through auction to the best available TTP who meets their service preferences.

The core of this architecture is layer-2 which resides between the vehicular drivers and SPs. This layer plays the role of the auction mediator service for the VDs, and it consists of four primary software components: Storage, participants QoE, computation, and the *auction manager*. The auction manager is the brain of this unit within the TTP layer at the cloud level. Storage unit is responsible for storing of all information of vehicle drivers and SPs which will be used later for the auction filtering and negotiation purposes. All of their ongoing information and related to provisioning services will be stored at this unit. The storage is also of paramount importance to formulate the database for the future selection of SPs based on their QoE reputation value for each requested service. Drivers have the choice also to choose auction or manual approach of selecting their TTPs. In case of manual selection, storage database is partially available to the VDs to browse through the list of available TTPs with their services-price matching. On the other hand, if they select an auction approach, then, they will be connected to auction manager who handles the auction on their behalf. Auction manager unit is directly connected to the computation and participants QoE units to fetch or process any required information. It manages, negotiate, and filter auction participants. The participants QoE unit is mainly responsible for storing the current QoE values of auction participants, namely the drivers or providers. This unit is concerned with the current auction QoE values only including suitable SPs for requested services. The computation unit handles all the computing tasks of the requested service whether manual or auction approach. This entity is similar to the auction manager; however, decoupling the communications and the computation enables more generalized design and less errors in run time. TTPs from the middle layer are aware of all cloud service provider's services and have their QoE values for each of their

services. Computation unit updates TTPs storage unit of any new available services with an existing service providers.

The SPs software part is not in the scope of this study. However, it is worthwhile mentioning that SPs usually offer different types of services such as the infrastructure, platform, storage, computation, applications and communication services as illustrated in Figure 5.2. Most of these services are supposed to be available with any cloud service provider.

5.4.2 AUCTION-DRIVEN QOE PROVISIONING (AQOEP) APPROACH

In this section, we explain the auction-driven QoE-based provisioning approach. We aim to incorporate the concept of auction-theory with our vehicular provisioning QoE framework. Auction approach contributions expected as follows: (1) Maximize SPs and TTPs profit. (2) Minimize {delay, cost, and IR} at the vehicle drivers side. QoE-aware at the TTPs side will be maximized using auction approach. Auction manager automatically assigns drivers with the best TTPs who potentially meet their interest in the requested services. In addition, auction manager selects only qualified SPs to participate in the auction. Moreover, when apply the concept of the auction theory to our QoE-based framework, every individual (Requester or Provider) has an added value (QoE feedback) to the auction. This will facilitate managing the auction and adapt participant's feedback. The following equations formulate the auction model based on the assumed contribution by each participant.

$$Pr_{SP}^S = P(\{TTP_i\}^S) - SC_S \quad (5.2)$$

$$Pr_{TTP}^i = P(\{VD\}^i) - SF(SP_S) \mid TTP_i \rightarrow SP_S \quad (5.3)$$

$$Pr_{VD}^x = QoE f(D, IR) - SF(TTP_i) \mid VD_x \rightarrow TTP_i \quad (5.4)$$

In the equations (5.2)-(5.4), Pr_x stands for profit of entity X where X can be a vehicle driver (VD), trusted third party (TTP) or service provider (SP). In the equations, $(\{TTP_i\}^S)$ stands for the TTPs that purchase service from SP_S . The profit denoted here is

the actual gain of services by the VDs or returned (payment/compensation) by the TTP/SP. $SF(X)$ denotes service fee payments made to entity X while SC stands for the service cost. Service fee is the amount paid to the provider (i.e. TTP, SP) from the requester (i.e. VD, TTP) on the required service. Similarly, service cost is all expenditures to deliver any service to any requester (i.e. TTP, VD). Also, $P(X)$ represents the payments received by entities in the set- X whereas C stands for the cost of service provisioning. It is worthwhile mentioning that, we define the request relation between two entities, X and Y such that $X \rightarrow Y$ stands for X requesting service from Y .

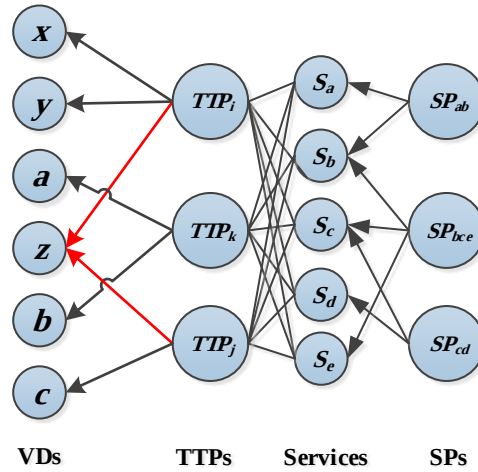


Figure 5.3: AQoEP selection procedure scenario

Auction manager is running all the time and mainly looking to find the set of VDs which would maximize TTPs income. Each TTP is managing one cluster with group of participants assigned to him from the auction. Each cluster setup a profit margin value based on what he already made and what he expects to gain for a particular service as shown in Eq. (5.5). The same concept can be applied to the SP to select the best set of TTPs where best set denotes the set of VDs that maximizes the income of the TTP and SP as shown in Eq. (5.6). Auction manager selects SPs who have the best QoE reputation value to join the auction for the requested service. The profit margin value of each cluster will be a key determining point to evaluate how many VD participants are need to reach an acceptable overall profit. The VDs in this approach will end up receiving the most satisfactory service no matter which service provider or TTPs are serving them.

$$Pr_{TTP}^{margin} = \left(\sum_{i=1}^n P(\{VD\}^i) - SF(SP_s) \right) + \sum P(\{VD\}^i)_n \quad (5.5)$$

$$Pr_{SP}^{margin} = \left(\sum_{i=1}^n P(\{TTP_i\}^s) + SC_s \right) + \sum P(\{TTP_i\}^s) \quad (5.6)$$

The selection or we may call auction winning procedure in AQoEP shown in Figure 5.3. Figure 5.3 overviews a simple scenario where three TTPs $\{i, k, j\}$ are to manage a simple auction of six VDs $\{x, y, z, a, b, c\}$. TTP_i can serve three participants $\{x, y, z\}$, TTP_k can serve $\{a, b\}$, and TTP_j can serve two participants $\{c, z\}$. On the other side of the auction, three different SPs with five different type of services are available for drivers. We have also proposed to run two auctions at the same time. First auction between the TTPs is to secure drivers with best service cost right up the front. Second auction between the SPs is to secure drivers with best QoE including less delay, less cost, less revealed information. Both auctions deliberate to get the drivers with best service cost at each service request, thus, the total service cost is expected to be reduced enormously. TTPs auction targeting cost only while SPs auction targeting entire QoE because TTPs are more trusted than service providers and their main objective is to achieve best QoE. However, the QoE framework has several TTPs providing the same service which implies a competitive environment between them to work out the best service. Once again, auction manager is aware of all SPs and their available services at all times. Running two auctions can be very complicated depending on the following factors: Number of drivers, type of the requested service, available trusted third parties and service providers to provide such services. To simplify this procedure, we present it in the following steps:

As seen from Figure 5.3 and selection procedure steps, upon the arrival of the new VD, the selection of the new VDs to join and form the TTP cluster arises as the first problem to address. In the scenario above, VD_z has been chosen and potentially can be served by two different TTPs (TTP_i and TTP_j). The following question may rise: How can one ensure that VD_z has been chosen by the most suitable TTP out of those two to receive their best services? On the other hand, how this will affect the other TTPs, and how to make sure it is in their best of interest to adopt this VD? Second, how should the pre-

existence of VDs be handled within the same cluster so that all VDs (previously added and newly arriving) are treated the same way?

Algorithm 1: Selection and Joining Procedure

Step-1: Each $TTP_{i...n}$ keeps current VDs ($\{VD_{x...y}\}$) on cluster with the same old service charges ($\{\{P_x\}\}$). Once both TTPs have VD_z service type, both auctions start.

Step-2: Auction 01: If TTP_i and TTP_j target the new arrival $\{VD_z\}$ who is willing to pay for best $QoE f(D, P, IR)$; (see Figure 5.3 for illustration)

Step-3: TTP_i and TTP_j set their profit margin according to Eq. (5.5) and start providing (bidding) VD_z discounts on the requested service but still greater than their marginal profit. The profit margin is the minimum service price that any player cannot go below it. Mainly, below this value, the providers may loss within his overall cluster profit.

Step-4: When both reaches to the minimum profit margin they could offer, TTP_i and TTP_j stop and decide based on what best final price to serve VD_z , e.g. TTP_j best fit for VD_z .

Step-5: TTP_j is bonded to VD_z with the most recent offered price.

Step-6: Auction 02: SPs who can provide requested service run Steps 2 to 4 until best SPs win the auction.

Step-7: TTP_j now can proceed with the service delivery between VD_z and best SP who won auction 02.

Step-8: After TTP_i wins the auction, he compares its profit margin with VD_{xy} and apply their charges to the new margin or offer them some compensation promotions.

Step-9: TTP_i increase his profit and all VD_{xyz} receives best $QoE f(D, P, IR)$ services.

In order to solve these problems, we propose that the auction should be based on the QoE-aware framework. It inherits all benefits of participant's feedback regarding provisioned services and use QoE reputation values as the input metric to this auction. These values play a major role of deciding which TTPs might participate in the auction or remain silent. The same rule applies on the vehicle drivers and service providers. In case if they have very bad overall QoE rating, then, they will not be qualified for the auction approach. The following sub-section presents a detailed step by step example of auction selecting and joining process.

5.4.3 AQoEP SELECTION AND JOINING

Figure 5.4 shows an example of AQoEP selection and joining process. We describe it in four main steps. Initially, VDs request to receive services using auction approach, if there QoE values are not qualified for auction approach, they will be re-directed to manual service selection. If qualified, they will join a general pool of auction participants as

shown in Figure 5.4 – step 1. Once they are in the general participant’s pool, auction manager becomes aware of their preferences and what type of services they are looking for. At this point, the auction manager commences two independent auctions: (1) between potential TTPs who are specialized with the requested services and able to serve driver’s request, and (2) between service providers who possess this service. We run the two auctions at the same time without waiting for the results of the other because their results are independent from each other and the long overall system delays should be avoided. The auction manager associates each VD in the pool with the appropriate TTPs who can serve those participants considering only the best TTPs who have the highest QoE for the auction. For instance, if there are five TTPs who can provide a VD with the requested services, then only the best three QoE will be chosen to join this auction. The minimum number of auction bidders (i.e. TTPs) should not be less than 3. This highlights another strong motivation for system participants to maintain their QoE in good standing. In the proposed example, TTP₁, TTP₃, and TTP₄ are qualified for the auction while TTP₂ has been excluded. Auction starts between those TTPs once the auction manager assigns VDs with the qualified TTPs as shown in Figure 5.4 – step 2.

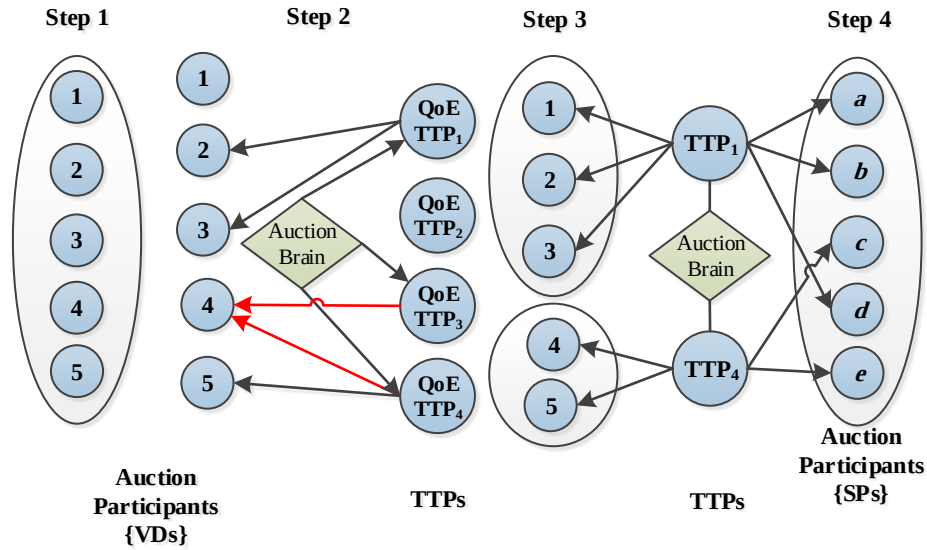


Figure 5.4: Example of AQoEP selection and joining

In this example, we have TTP₃ and TTP₄ trying to win the auction of VD₄ and bring VD₄ to their clusters. TTP₃ and TTP₄ are trying to maximize their profit and at the same time

guaranteeing the delivery of the requested services to VD₄. They have to calculate their current marginal profit using Eq. (5.5), and use it with this auction. The equation considers SPs service fees for any individual service. TTPs anticipate service fee based on the last recorded charges for similar services. Steps 1-7 defined above will be executed until a TTP offers a service charge less than the other TTPs charges. The marginal profit of each cluster is a dynamic value that changes based on the type of the requested services and the number of the cluster participants. The higher number of participants the better marginal profit can be negotiated. In this example, TTP₄ provides better marginal profit, won the auction, and VD₄ joins TTP₄ cluster along with his other participants as illustrated in Figure 5.4 – step 3.

On the other side of the auction, auction manager uses the same concept and qualifies the best SPs based on SPs QoE reputation values to join an auction between different SPs to provide a particular driver with the requested service. SPs also use the same concept and Eq. (5.6) to find their “best” sets of TTPs who expect to purchase their services to increase their profit. Thus, a SP who qualified for the auction have the chance to decline the auction manager’s offer in terms of the requested services. However, SPs are committed to deliver drivers requested service once they win auction and associated with service request as shown Figure 5.4 – step 4. Following this approach, all participants are expected to maximize their profit, auction QoE becomes more informative which has a positive impact on future auctions. Such approach requires that the TTPs storage of participants QoE should be high, in other words, the more experience in the system, the auction better the auction performance.

One important point to consider here is that a greedy TTP may be able to win more auctions and collect more profits than other TTPs or SPs if they consider to lower their marginal profit. However, this may affect their efficiency, and it will be reflected negatively on their QoE reputation. Once QoE reputation drop down, the auction manager starts excluding them from incoming auctions. Thus, the number of participants at each cluster should be within the limit of the TTP capacity to avoid any service disruption or delay. The AQoEP scheme provides many benefits to our system participants; however, the following questions should be answered: Will the number of

the running auctions increase overall system delay? What is the chance of an auction participant to win no auction? The following section presents the performance evaluation of the proposed auction scheme and present answers to these questions.

5.5 PERFORMANCE EVALUATION

In this section we evaluate and compare the following approaches: *Naïve QoE-based Provisioning (NQoEP)*, *Neutral mode*, *Sensitive modes*, and *Auction-driven Quality of Experience-based Provisioning (AQoEP)* in a vehicular cloud. The purpose of evaluating these approaches is to investigate the worthiness of using auction theory with QoE, to determine how auction will affect driver's satisfaction of privacy, service latency, and service cost, and measure the level of complexity of auction approach with proposed QoE framework.

5.5.1 SIMULATION SETTINGS

Table 5.1: AQoEP simulation scenarios

<i>Scenarios</i>	<i>Vehicles</i>	<i>TTPs</i>	<i>Service Providers</i>	<i>Sensitive modes</i>		<i>AQoEP Mode 1</i>	<i>AQoEP Mode 2</i>
1	5	2, 3	3, 4	Accordingly	$a = 1,$	QoE-aware Coefficients $a, \beta, \gamma = \text{static} = .33, .33, .34$	QoE-aware Coefficients $a, \beta, \gamma = \text{random.}$
2	10	3, 5, 6	5, 7, 7		$\beta = 1,$		
3	15	5, 9, 9	9, 12, 13		and		
4	20	9, 11, 15	10, 13, 17				
5	25	14, 16, 19	18, 20, 22		$\gamma = 1.$		

In the following simulations, we used Network Simulator 3 (NS-3) instead of NS-2 to collect the numerical results of the proposed approaches. However, we re-used the exact same communication settings used with NS-2 in previous chapter. These evaluations inherit the same communication medium and mobility model presented in 4.4.1 and 4.4.2, respectively. In the upcoming set of simulations, each was repeated 10 times with a range of 5 to 25 vehicles at each time. Several scenarios of testing have been applied as well. The aim of this testing is to find the level of complexity of the auction procedure under different network densities and determine the impact of the corresponding

complexity on provisioning services with mobile network. We have re-tested the aforementioned sensitivities modes: *Delay-sensitive*, *Economy-sensitive*, and *Privacy-sensitive* under the auction approach. The definitions of these terms have been presented in CHAPTER 3 and CHAPTER 4. AQoEP has been also deployed in two modes: random and static QoE coefficients as shown in Table 5.1.

5.5.2 SIMULATION RESULTS

We have evaluated the proposed approaches in terms of the following metrics:

1. *Delay* is the end-to-end time before a driver receives the requested service from the service provider. This metric is used to compare the delay generated by auction model and other conventional models.
2. *Price* is the total that drivers are charged based on their usage, applied by the trusted third party and the service provider. A *saving* price parameter is evaluated with this metric, to represent how much drivers will save if they adopt one approach over another.
3. *Privacy* is the amount of information revealed to the service provider. It represents the actual personal information a driver must provide to the service provider through the trusted third party to receive the service. The information required will vary for different services.
4. *Level of complexity* which measures how long it takes for an auction to be stabilized in terms of existing participants and newly joining ones. This metric considers how long it takes participants to spend in the auction general pool. This metric applies to both pools of vehicle drivers and service providers. This metric is different than delay metric. Delay metric is about the total time before participants receive the requested service while level of complexity is about the number of auction stages.

In the first group of the simulations, we evaluated the performance of the average delay under the proposed different approaches as shown in Figure 5.5. *Delay-sensitive* mode produces the best average delay in the early dense of the network (5-15 vehicles) while

Neutral mode produces the poorest delay comparing to *Delay-sensitive* and *NQoEP*. The alpha value (α) of the *Delay-sensitive* mode in equation 5.1 is set to one, the reason behind such efficiency. Drivers interested in minimizing provisioning service latency can adopt this mode especially if the type of the service can affect critically by the delay such as multimedia streaming content. Figure 5.5 also shows that both modes (M1: static and M2: random) of AQoEP present a constant gap of ~15% of delay in 5 to 20 network density. This gap increased to about 20% in the largest network density (25 vehicles). When more vehicles are included in the simulation, the auction-based provisioning improves the system in terms of experience and feedback. In addition, at this density, AQOEP-M2 shows a slight difference in the delay produced when compared to *Delay-sensitive* mode. This difference has been reduced gradually starting from 5 to 25 vehicles. Moreover, AQoEP-M2 outperforms NQoEP which results in producing less delay than NQoEP in all network densities. The concluding remarks about the average delay performance in the auction modes are, AQoEP offers better readings (less delay) when the number of the vehicles is increased, AQoEP-M2 proves that on the long run random coefficients factors better than static ones, *Delay-sensitive* mode is recommended for special type of services affected a lot by delay such as real-time applications, and NQoEP still has less average delay than naïve modes such as *Neutral mode* because of the experience and driver's feedback (rating).

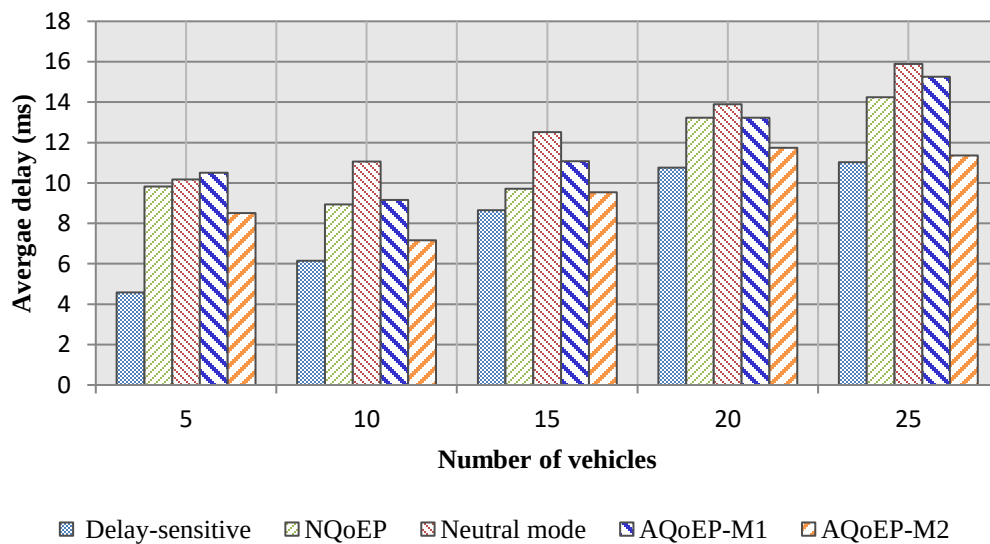


Figure 5.5: Delay performance

In the second group of the simulations, we compare the service cost performance under *Neutral*, *Economy-sensitive*, *NQoEP*, and *AQoEP (M1 and M2)* modes. Figure 5.6 clearly shows that *Neutral mode* is entirely the worse to consider since all other approaches have supported the driver with less average service cost at each single network dense. The value of beta (β) in Eq. (5.1) is set to one, the reason behind such efficiency. However, it is not the case when it is compared to the *AQoEP* modes. The remarkable enhancement in this group of simulations is that both modes of *AQoEP (M1 and M2)* outperform the *Economy-sensitive* mode whereas *AQoEP* always provides lower average service cost. This is the outcome of the advantage of using two independent auctions to negotiate the service charges of both TTPs and SPs. Both auctions have excreted the best TTPs with lower charge rates and the best SPs with the highest QoE. Hence, it is more viable to adopt *AQoEP* than manual third parties' selection from the user-centric point of view. Moreover, *AQoEP-M2* proves that random coefficients factor better to use than static ones whereas the service charges in random mode reduced by $\sim 20\%$ with the increase of the network density.

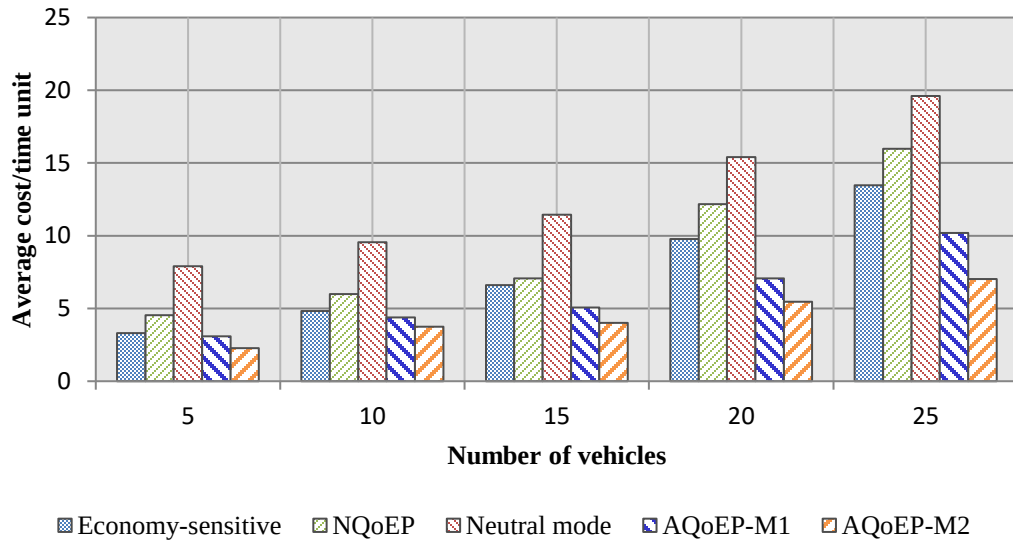


Figure 5.6: Service cost performance

In the third group of the simulations, we evaluated the performance of the average revealed information under the proposed different approaches. Figure 5.7 shows the enhancement on the amount of the revealed information to service providers as a necessity to deliver services to drivers. *Neutral mode* is outperformed by the other modes

as it does not have any knowledge on how much and/or what type of information is needed to be provided. On the other hand, *Privacy-sensitive* mode has already proven its feasibility in terms of privacy. The amount of revealed information under the *Privacy-sensitive* mode still less than that is revealed under the *NQoEP* mode. However, it shows very similar results in the first 4 network densities when compared to the *AQoEP* modes. Only the last network density leads to a lesser information reveal. The auction also demonstrates better performance in case of random coefficients since *AQoEP-M2* is slightly better than *AQoEP-M1* as shown in Figure 5.7. *AQoEP-M2* has shown higher improvements in delay and cost in comparison to the revealed information.

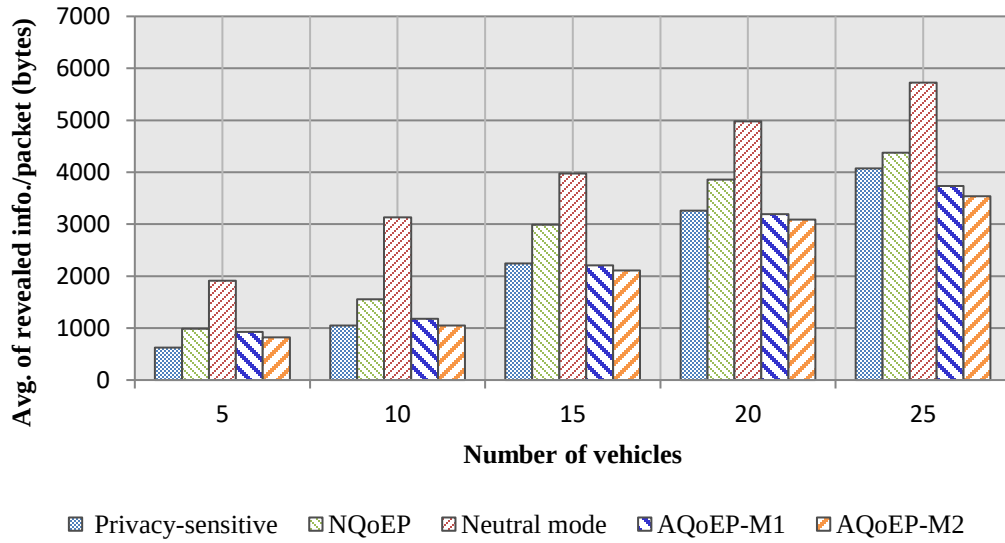


Figure 5.7: Revealed information performance

In the last group of the simulations, we evaluated the level of complexity under the proposed *AQoEP* modes. At this set of simulations, we increased the network density to 50 vehicles and tested the durations of the TTPs and SPs auctions to deliver requested services to drivers. Figure 5.8 shows that *AQoEP-M2* takes longer time than *AQoEP-M1* to stabilize. The coefficients play a role in the QoE auction formation which render the random coefficients taking longer time than the static ones. The overall difference between the two modes is not significant at the first 4 groups of density and the gap increased slowly in the last 6 group of density. We can notice that the auction complexity is linear depending on the number of the vehicles. The type of the requested service as

well as a more recent mobility model have not been considered in this simulation which may also have an impact on the complexity level. Those two factors have not been considered at this point of the research, however, an extended version of this work with the use of a suitable mobility model with different type of services is under investigation.

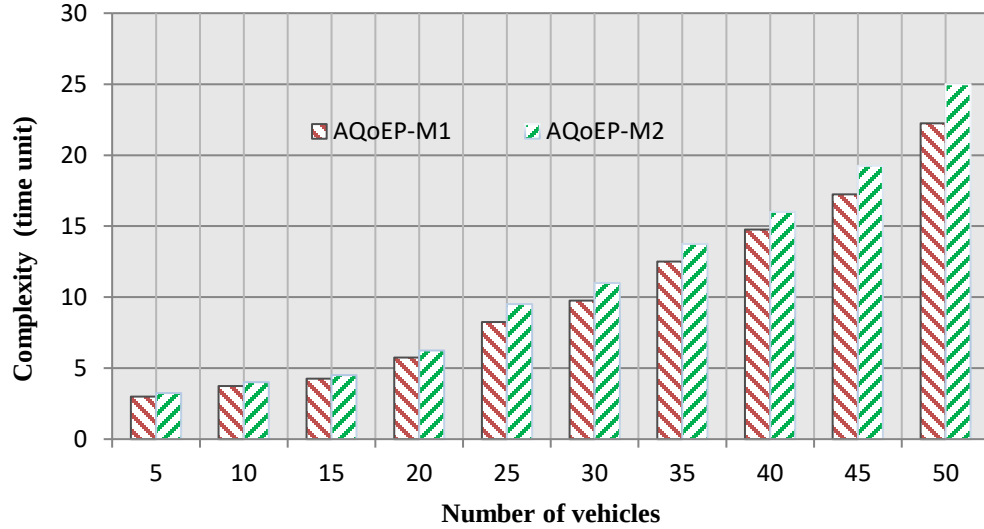


Figure 5.8: Level of complexity under AQoEP modes

5.6 SUMMARY

In this chapter, a new approach based on auction theory and quality of experience has been developed, named *Auction-driven Quality of Experience Provisioning (AQoEP)*. The proposed approach is compared to the QoE approaches presented in CHAPTER 4, both of which are commonly used to measure the service latency, service cost, and privacy in vehicular clouds. It is shown that the use of auction approach provides a better result in terms of service latency and amount of revealed information with a significant service cost reduction. Furthermore, when compared to the sensitive modes of drivers' preferences, the AQoEP approach is a remarkably more robust.

The characteristics of auction approach have been studied in both static and random modes using simulations. Result reassures the benefits of auction approach and adopting random coefficients of QoE. The service cost and the revealed information have been significantly reduced when the number of the vehicles increases. Further reduction in delay is also obtained as the number of vehicles increases. The auction has been shown to

be very close to Delay-sensitive and Privacy-sensitive service provisioning modes while it outperforms the economy sensitive approach in larger network dense. Therefore, it is in the vehicle driver's benefits to adopt AQoEP random mode over all other modes. However, the only drawback in such model is the level of the complexity since the auction has shown nearly linear growth when the number of the vehicles increases.

CHAPTER 6 AN INTERACTION GAME SYSTEM FOR SERVICE PROVISIONING IN VEHICULAR CLOUDS

6.1 INTRODUCTION

An auction-driven, multi-objective provisioning framework, with the support of the existing trusted third party approach and the Quality-of-Experience (QoE) model, has been proposed to address vehicular cloud challenges presented in previous chapters [127]. Auction interactions require an $n \times n \times n$ auction model, and the drawbacks of these models include delay and auction complexity. The delay has a significant impact on service charges, since drivers are charged per usage duration.

In this chapter, we propose using game theory concepts with a *QoE-aware* system model, to provide drivers with provisioning services in a vehicular environment at low latency, minimum cost and minimal driver information revealed. *QoE-aware* collects requirements and preferences, and defines a QoE value for every service provider, trusted third party and driver in each game. Three different games have been developed namely, Game n, Game a and MMIGS game. Game n and Game a are simple games without any QoE features between its players, however, *Game a* are similar to the former auction-based framework presented in CHAPTER 5. Multiagent/Multiobjective Interaction Game System (MMIGS) are more interactive game with two possible levels of interaction, and negotiations between players. We evaluate the proposed games in different network scenarios, with respect to driver privacy, service cost and latency. The results show that the proposed framework provides improvements over other conventional models in terms of these metrics. Service cost and privacy have been improved by 65% and 47%, respectively. Though these improvements are at the expense of extra delay, using QoE-aware has helped reduce such delays an average of 3%, compared to the total delay in other models.

The balance of the chapter is organized as follows. Section 6.2 presents the game-based of the QoE framework. Section 6.2.6.3 and 6.4 describe the proposed game system

models and analysis of the outcomes, respectively. Section 6.5 provides performance evaluation and simulation results. Finally, Section 6.6 presents the conclusions and discusses future directions.

6.2 GAME-BASED OF THE QoE FRAMEWORK

In the previous chapter, we developed an auction-based, multi-objective framework for service provision in vehicular clouds [127]. The multi-objective framework focuses on cost, latency and privacy, and it has shown promise since drivers are dynamically bound to the best fit available from a trusted third party. In addition, the trusted third parties can maximize their income by selecting the best cluster of drivers. This solution can be applied and still be valid for a one level (two end) auction; that is, the buyer (driver) and the seller (trusted third party) at either end. Our one level (two end) auction has $n \times m$ participants, where n represents the number of drivers and m represents the trusted third parties. For example, at any time a number of drivers could be bidding on several services through different trusted third parties.

The proposed auction includes three main entities (i.e. drivers, trusted third parties and service providers), and at some point the trusted third parties must play the roles of buyer and seller at the same time; buying from the service providers and selling to the drivers. Such an auction-based solution assumes that the trusted third parties will provide the drivers the promised services, and ignore the second level interactions between the trusted third parties and the service providers. Moreover, some drivers will be unable to bind to any available trusted third party, if none are willing to bind with them.

To address this issue and find the best cluster match (i.e. drivers with trusted third parties, and trusted third parties with service providers) for a single or multi-relationship auction, we ran a two-level (three end) auction with buyers (drivers and trusted third parties) and sellers (trusted third parties and service providers). These interactions require an $n \times m \times r$ auction model, where n represents the number of drivers, m represents the number of trusted third parties and r represents the number of service providers. The drawbacks of this model are the delay, the auction complexity, and the negative impact that the delay has on the service charges, since the drivers are charged by time used. These types of

solutions can only be adopted in small network dense and the delay is monitored and evaluated correctly. Moreover, an $n \times m \times r$ auction is very difficult to accomplish because the relationships can be different at each level, which translates to computational complexity and a need for high processing power. The outcomes can vary and often be unexpected.

Related work manages such problems with multi-agent game theory systems [128][129][130]. A description of possible solutions using different game model approaches is presented in this section.

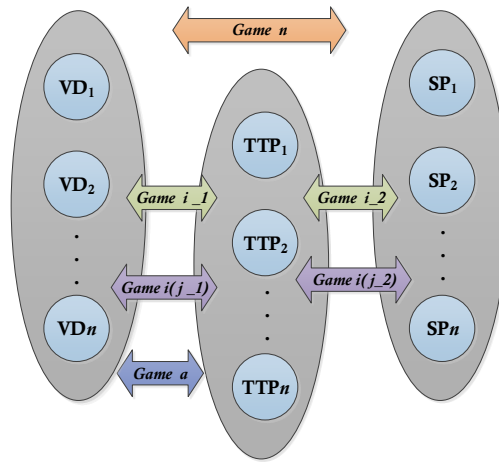


Figure 6.1: Game-based of the auction framework

To prove our concept using game theory we developed three different games, as shown in Figure 6.1. These are: *Game n*: a naive game between drivers and service providers; *Game i* and *j*: a Multiagent/Multiobjectives Interaction Game System (MMIGS) among all system participants; and *Game a*: an auction game between the drivers and trusted third parties.

6.2.1 GAME DESCRIPTIONS

Game n is a naïve simple game between two players, namely the drivers and the service providers. It does not consider latency, cost or privacy, and does not introduce any QoE awareness between the players. A driver plays the game with several service providers, and one of the providers wins the game and provides the requested service to the driver.

Game i and *j* is a multiagent/multiobjective interaction game system with two possible levels of interaction, and negotiations between three main players: the drivers, trusted third parties and the service providers. This game ensures the driver will get the best price, minimal latency and enhanced privacy. In addition, the trusted third parties and the service providers play the game to maximize their returns, which in turn translates to a win-win situation. MMIGS adopts the concept of the QoE awareness at each level of the game.

Game a is a sightless game between drivers and trusted third parties. It does not involve service providers, or implement any QoE awareness between the players. Trusted third parties are appointed by drivers to ensure they get basic QoS requirements, such as latency and cost. This is analogous to a situation where two players gamble without the knowledge of the possible outcomes. Some aspects of *Game a* are similar to the former auction-based framework we presented in [127].

6.2.2 OBJECTIVES

The objectives of these proposed games are as follows: 1) to provide a comprehensive study of using the game model with different approaches; 2) to prove our concept of using the QoE framework with game-based; 3) to meet our proposed objectives for service provisioning in vehicular clouds; 4) to investigate the suitability of a game theory-based approach rather than an auction-based approach for our proposed QoE-based framework; and 5) to prove that using MMIGS is the best fit solution for service provisioning.

6.3 SYSTEM MODEL

By nature, game theory introduces beneficial intermediate results for all players [131], which is why we chose this approach and integrated it with our QoE framework. Game theory approach has been used extensively with several research bodies. For example, the authors in [129] propose an algorithm for cooperation among roadside units in a vehicular network using coalitions game. Another [132] has explore how to adapt Nash Equilibria for the service provisioning in cloud systems. Less effort has been devoted to

explore the use of game approaches with service provisioning in vehicular clouds. Integrating the game theory-based approach with our QoE framework guarantees mutual benefits for all participants, and it is a more efficient system than other existing models. The system ensures that drivers get satisfactory service, and services providers (i.e. trusted third party and service providers) receive adequate profit. The following are explanations of the different theoretic approaches used for the proposed games [133].

6.3.1 “GAME N” MODEL

Game n is played between two players who have previously shared common knowledge about each other (e.g. utility charges, reputation, expected delay) without QoE awareness. It assumes the players are engaged in the game simultaneously, as in our previously proposed *Neutral mode* in [127]. However, the *Game n* model is built on a theoretical game approach, which was not the case with the previously presented neutral mode. The QoE-based approach is not included in this model, as shown in Figure 6.2.

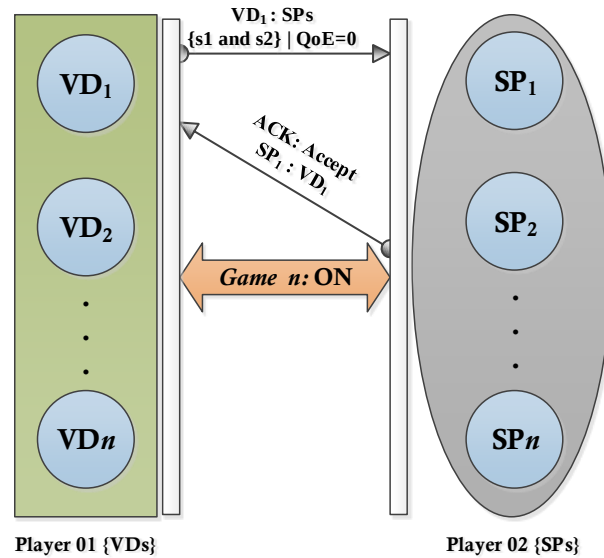


Figure 6.2: *Game n* approach

Game n allows player 01 (i.e. a driver) to start the game, and they have all available information about player 02 (i.e. a service providers). Similarly, player 02 is given access to all information about player 01. For example, at some point VD_1 seeks services $\{s_1 \text{ and } s_2\} \in \mathcal{S}$ from service providers, without any conditions such as latency or

privacy. The service providers accept or decline a request based on the information about VD_1 and the nature of the request. The details of a driver's request (e.g. time, type) play a role in the decision to accept or decline the request; if the service provider accepts it the game begins between them.

The outcomes of *Game n* always favour player 02. Even if the players have common knowledge about each other and expectations of utilities and latency, player 02 can select in their best interest before approving a driver's request. Thus, this model does not guarantee that player 02 decisions are mutually beneficial. In addition, player 01 has no control over their data, nor any guarantees about how the personal information will be used (i.e. absence of QoE). Moreover, the lack of QoE awareness between drivers and service providers means service providers can play in their best of interest only. The drivers have no information or knowledge about service providers' quality of service (e.g. service latency, service cost), and there is no vehicular services feedback from prior drivers regarding the requested services. Nonetheless, service providers tend to offer their best service in order to protect their reputation.

6.3.2 MMIGS MODEL

The Multiagent/Multiobjective Interaction Game System (MMIGS) has two potential levels (i.e. i and j) of interaction between drivers (player 01), trusted third parties (player 02) and service providers (player 03). Both the QoE-based framework and the theoretic game approach are adopted (i.e. the Nash Equilibrium) to provide players' experience throughout the game, and to introduce multi-objective services among multiple agents.

Four main games (i_1 , i_2 , $i(j_1)$ and $i(j_2)$) can be applied during the game. The participants' strategy is to play their best response between each other during the game. Put simply, player 01 (drivers) aim to get the best services from trusted third parties and service providers, while players 02 and 03 want adequate return when handling or providing services. Three main service objectives (i.e. latency, cost and privacy) are managed during the game, and are the building blocks of our QoE-based framework.

A) MMIGS GAME I

Game (i_1) begins when a driver initiates a service request to a specific trusted third party, and indicates their preferences about the requested service. This means that drivers must provide their best play options, since they know that the QoE-based framework will select the player with the best available services, according to the preferences. The drivers' QoE-based model offers the option to review the trusted third parties and decide which to consider, based on the player's current interest. The MMIGS model implies that drivers and service providers will give the trusted third party their preference values (i.e. $\{D, P, I\}$) for each requested service, as stated in Eq. (5.1).

Once the trusted third party has a driver's requests, they run the game on the driver's behalf. The trusted third party has access to the driver's preferences, and plays its part of the game without disclosing any of the drivers' information or interests. Trusted third parties have more QoE knowledge about service providers than individual drivers. The second part of the game (i_2) is initiated by the trusted third party, who provides the driver's request to the service providers that can match them. Each service provider knows that this game has more than two parties and more than one service provider, which motivates them to play their best choices, not only the winning choice as in *Game n*. Service providers' responses are rated after the services have been delivered to the driver, and the ratings could have a positive or negative impact on a provider's overall QoE reputation. The trusted third party receives the service providers' best offer (i.e. $SP_j \{D_j, P_j, I_j\}$) for the driver's requested services. The trusted third party then constructs a matrix of the service providers' responses (SP_j) to the driver's original request.

The trusted third party then examines the offers and compares them to the driver's request, after which the trusted third party accepts the best offer and notifies all parties accordingly. The choice is based on the trusted third party acting in the best interest of all parties, not only for itself. Thus, the trusted third party's best strategy is to find the best match for the driver's request, select it, add their fee, and notify all parties of the outcome. The trusted third party's duty is to find drivers the best services that match their preferences, and this is not necessarily straightforward. It becomes more complicated if

none of the service providers have extended an offer that matches the driver's request, which means more interactions between the multiagents is required during the game.

B) MMIGS GAME I(J)

MMIGS game i should be more resilient against unexpected situations, which requires more negotiation between game participants about available choices. More interaction between the parties ensures that all possible options are available to all players, to help them reach a more optimal game model. Though more communications typically lead to additional delays, reasonable extra delays can sometimes have a positive effect on the overall outcome of the game. *MMIGS game i(j)* uses the *game i* level, with an extension to provide more negotiation on the available choices.

MMIGS game i(j) can only be activated if drivers and service providers are open to negotiations, re-visiting and amending the terms of their offers, and the trusted third party feels there is a good chance to start negotiations.

The extension *game i(j)* of *game i* can be only possible if the $SP_j(\mathcal{S}) = \sum_{j=1}^n \{(D_j^s, P_j^s, I_j^s)\}$ matrix and $VD_i(\mathcal{S}) = \sum_{i=1}^n \{(D_i^s, P_i^s, I_i^s)\}$ are not considered a suitable match by the trusted third party, they are very close to each other, and none of the terms compromise others. For example, given that a driver is willing to tolerate some extra delay, and has more privacy information or less to pay, the trusted third party runs the closest options from the generated matrix on the driver. The driver studies the offers and decides to either accept one, or propose a new request with the amended values of the new offers. If the driver accepts one of the closest offers there is no need to notify the service providers, since one of the objectives is to reduce the latency. However, if the driver proposes a new game request (i.e. *game i(j_1)*) with valid enhancements of the initial preference values, the appointed trusted third party continues the game and passes this new request (i.e. *game i(j_2)*) to the group of service providers involved. Participants can interact with each other to help make the negotiations successful, and it is assumed that they will agree that any new proposed offers must have more persuasive compromise values than before.

At this point, the trusted third party has received the amended offers that satisfy the driver's new proposed offer from the service providers. Finally, the trusted third party passes the amended offers from the group of service providers to the vehicle driver. If the driver accepts them, an acknowledgment message is sent to the chosen service provider permitting delivery of the service.

At the end of the game, and once the driver has received the service(s), all participants can rate both the services and the game. Drivers typically rate the service they received based on their level of satisfaction, while trusted third parties rate the drivers and the service providers, based on the interactions and the level of the promises made between them. Service providers can also rate their interaction with the trusted third parties during the game.

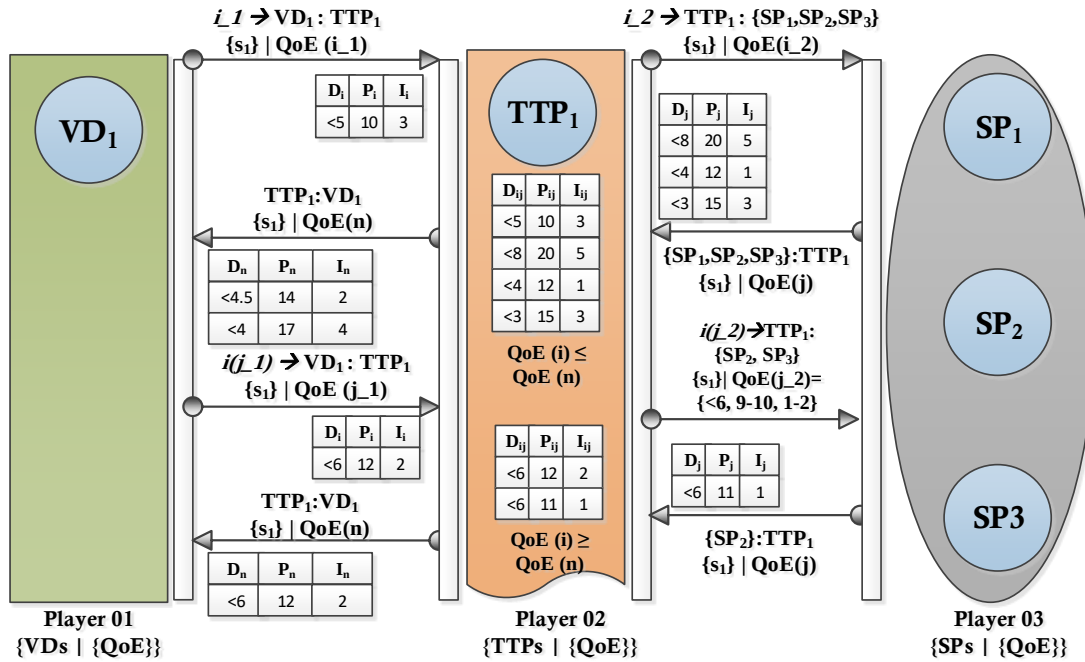


Figure 6.3: An example scenario of the MMIGS game approach

C) MMIGS GAME I(J) SCENARIO

Figure 6.3 illustrates game i and game i(j) in detail. At the start, VD₁ is seeking services $\{s_1 \in \mathcal{S}\}$, and selects TTP₁ to play the game on their behalf. VD₁ chooses TTP₁ because it is the best match for the service and interests of VD₁. Drivers can select any other

available trusted third party, based on their QoE reputation. The QoE for the participants are then announced and are available for all parties on the cloud, which is why each party in our QoE-based framework wants to maintain their QoE reputation. In most cases, selecting the corresponding party in the QoE framework is based on the QoE reputation about a particular service.

A one-to-one game (i_1) starts when TTP_1 receives the VD_1 request and all the related request information:

$$QoE (VD_i) = \{D_i, P_i, I_i\} = \{<5, 10, 3\} \quad (6.1)$$

TTP_1 keeps client (e.g. VD_1) information protected, and applies its previously found knowledge and experience to find the most suitable and available service providers (e.g. SP_1, SP_2, SP_3) to contact about the VD_1 service request. A one-to-many game (i_2) is initiated by a TTP_1 request to the service providers with the appropriate QoE reputation to provide the services. The intention of TTP_1 at this stage of the game is to release only the service type $\{s_1\}$ to the service providers, and hide the driver service preferences (e.g. $\{<5, 10, 3\}$). At this point, TTP_1 has no information about offers from service providers for the service, so the best strategy is not to disclose the driver preferences to the service providers.

The service providers know that there are other service providers in the game, so they will respond with their best offer. This keeps them in the game, and gives them a significantly better chance of winning than playing for their own benefit and being excluded from the game later. Thus, as in Figure 6.3, the three service providers have responded with the following offers:

$$\begin{aligned} QoE (SP_j) &= \{D_j, P_j, I_j\} \\ &= \{ \{<8, 20, 5\}, \{<4, 12, 1\}, \{<3, 15, 3\} \} \end{aligned} \quad (6.2)$$

Once TTP_1 has all the service providers' offers, it selects the most suitable ones and puts them and the VD_1 preferences into a matrix to match and compare them, as shown in

Figure 6.4. If more than one service is requested by the drivers, match/compare matrices are built for each service.

	D_{ij}	P_{ij}	I_{ij}
VD_1	<5	10	3
SP_1	<8	20	5
SP_2	<4	12	1
SP_3	<3	15	3

Figure 6.4: QoE (i,j) preference comparison

TTP₁ finds that SP₂ and SP₃ are proposing the most appropriate offers. Based on this, TTP₁ determines its best option and responds back to VD₁ with a new matrix, as shown in Figure 6.5. The matrix is based on the most suitable offers proposed by SP₂ and SP₃ that were amended to make them suitable for TTP₁.

$$\begin{aligned}
 \text{QoE}(TTP_n) &= \{D_n, P_n, I_n\} \\
 &= \{ \{<4.5, 14, 2\}, \{<4, 17, 4\} \}
 \end{aligned} \tag{6.3}$$

	D_n	P_n	I_n
1	<4.5	14	2
2	<4	17	4

Figure 6.5: QoE (n) proposed offer to vehicle drivers'

One objective of trusted third parties is to spend more time and computational effort to protect drivers' personal information, while getting them the best available offers and services from the service providers. Thus, an acceptable offer to the drivers has better, equal or QoE values that meet their requirements. The semi-final proposed game offer from TTP₁ to VD₁ typically has the following priorities:

$$\{D_n, P_n, I_n\} > \{D_i, P_i, I_i\} \rightarrow \text{QoE}(TTP_n) > \text{QoE}(VD_i) \tag{6.4}$$

A typical game offer that can be considered by the vehicle drivers should meet the following:

$$QoE = QoE(VD_i) - QoE(TTP_n) \geq 0 \quad (6.5)$$

In this example, the QoE final offer shown in Figure 6.5, is not equal or less than what VD_1 is expecting. However, it is close, and persuasive enough to be re-considered for negotiation. At this point, VD_1 is not fully satisfied with the two offers because they cost more, even though they have less delay than specified, so, now the driver's best strategy is to re-negotiate the offers that have just been received. A new game ($i(j_1)$) with an amended offer is initiated by VD_1 to TTP_1 , stating that the driver is willing to pay slightly more (10 to 12) and experience slightly higher delay (<5 to <6) to get better privacy (3 to 2) for the personal information revealed.

TTP_1 receives the new VD_1 preferences, checks to make sure that VD_1 has made reasonable amendments to the extended proposal, informs SP_2 and SP_3 that a new amended proposal has been made due to driver dissatisfaction, starts a new game session ($i(j_2)$) with SP_2 and SP_3 , and passes the extended proposal directly to them. Thus far, TTP_1 has proposed two main games (i.e. *Game* (i_2) and $i(j_2)$), as shown in Figure 6.3. The games are not similar, since they are fundamentally different in terms of strategy. As explained earlier, since TTP_1 is not certain what it is negotiating against, its best approach for the first game (i_2) is to not disclose its driver's preferences to the service providers. However, in the second game ($i(j_2)$), TTP_1 reveals more of the driver's preferences, to show the service providers that the driver is determined to get this service. Another reason for revealing more is to give the service providers additional background about what they should expect. Moreover, at game ($i(j_2)$), the drivers' preferences are adjusted slightly, then passed to a small subset of the original service providers who have already responded with reasonable offers (SP_2 and SP_3). In our scenario, TTP_1 shares its client's preferences with only SP_2 and SP_3 , excluding SP_1 since its offer was not reasonable. TTP_1 then adjusts VD_1 's preferences, and introduces them to SP_2 and SP_3 as follows.

$$\begin{aligned} QoE(VD_{i_2}) &= \{D_{i_2}, P_{i_2}, I_{i_2}\} \\ &= \{<6, 9-10, 1-2\} \end{aligned} \quad (6.6)$$

SP_2 and SP_3 receive TTP_1 's adjusted preferences, examine them, and then play their best strategies. Service providers can quit the negotiations for any reason, although it could

affect their QoE reputation, depending on how often they quit games. In this example, SP_2 is the most likely candidate since its choices are very close to VD_1 's adjusted preferences, so its best play is to propose a new offer that fits the VD_1 adjusted preferences. Accordingly, SP_2 proposes a new offer ($\{<6, 11, 1\}$) to TTP_1 , as shown in Figure 6.3. TTP_1 receives the offer, then plays its best game option ($\{<6, 12, 2\}$) and matches its driver's needs at the same time. The final offer is delivered to VD_1 which is found to be a perfect fit, and is accepted.

$$QoE = QoE(VD_i) - QoE(TTP_n) = 0 \quad (6.7)$$

6.3.3 “GAME A” MODEL

Game a is a blind game between drivers and trusted third parties. Service providers are not involved, and there is no QoE awareness between the players. *Game a* begins when a driver requests to be bound to an available trusted third party; it is the driver's responsibility to include any further information in the request. If a trusted third party accepts the request, the driver informs them of the service(s) they want. A blind request with no information about the service type always favours the driver, for two reasons: the driver can refuse the trusted third parties offer, or they can choose to quit the game at any time if they receive an offer from one trusted third party while another is still proposing. However, this could affect the willingness of the trusted third parties to accept blind requests. If a trusted third party chooses to accept, then the game is played between them and the driver, as shown in Figure 6.6. Trusted third parties are supposed to guarantee basic QoS for their drivers, such as low latency and optimal privacy. However, it is a blind game, since the trusted third parties have no idea of the service type or cost. Another drawback of this game is if a trusted third party receives offers from service providers about a requested service, proposes them to the driver and the driver does not accept them. Thus, both players have no knowledge of the outcome. A scenario of this approach is illustrated in Figure 6.6.

VD_1 sends a blind request to trusted third parties to inform to start a game. TTP_1 and TTP_2 accept the request, which triggers VD_1 to send them the service type $\{S_x\}$. TTP_1

and TTP_2 contact the service providers to get their offers for the service $\{S_x\}$. Two service providers can provide the service, and they make their offers to TTP_2 . Meanwhile, another provider (SP_3) also makes an offer to TTP_1 . TTP_2 selects the best of the two offers received, adds their service fees, and proposes the final offer to VD_1 . If VD_1 chooses to accept the offer, they inform TTP_2 and quit the game with TTP_1 . The driver can also wait to see the possible final offer from TTP_1 , compare it with the TTP_2 offer, and select their preference. This scenario demonstrates that trusted third parties are more vulnerable in this game than drivers and service providers.

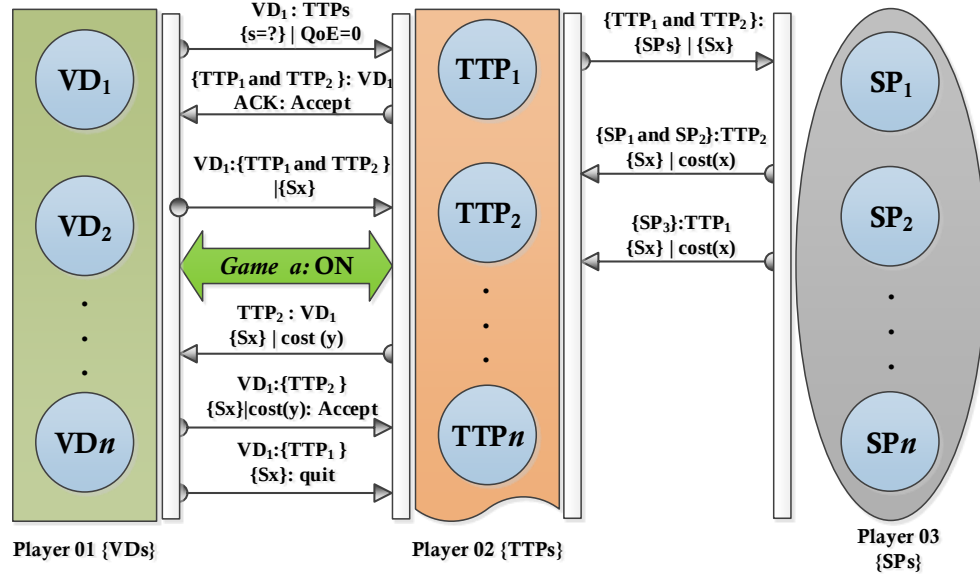


Figure 6.6: *Game a* approach

6.4 ANALYSIS OF OUTCOMES UNDER MMIGS

In order to understand the multiagent/multiobjective interaction game system, the types and roles of the players must be defined, as well as the possible interactions between them. In our proposed game system, we consider three main players $\{VD_i, TTP_n, \text{ and } SP_j\}$, each of which is independent and has preferences that represent their best interests. The expected outcomes can be defined by the following set:

$$\Omega_o = \{QoE_{VD}, QoE_{TTP}, QoE_{SP}\} \quad (6.8)$$

A set is made up of all possible outcomes of the game, and each outcome involves three different preferences (objectives) for each player. The set of these preferences is then defined (the preferences refer to the QoE factors in the QoE framework).

$$\Omega_p = \{D, P, I\} \quad (6.9)$$

Every player in the game has their own QoE, as shown in the outcomes set (9), and each wants to keep their QoE in good standing (the higher the QoE value the better). Thus, the system experience preferences function can be defined as follows:

$$QoE: \Omega_o (\Omega_p) \rightarrow \mathbb{R} \quad (6.10)$$

Our system has more than one possible outcome at a time for each preference. For example, if a and b are possible outcomes (e.g. a for delay, b for price) for QoE_{vd1} , and $QoE_{vd1}(a) \geq QoE_{vd1}(b)$, then VD_1 prefers outcome a over outcome b . Based on this, we can introduce a more general concept: if the possible outcomes for a player's preference are a , b and c , and given $a \geq b$ and $b \geq c$, then,

$$QoE(a) \geq QoE(b) \text{ and } QoE(b) \geq QoE(c), \rightarrow QoE(a) \geq QoE(c) \quad (6.11)$$

With this concept of player preferences, we can show the interactions between our players for specific or multiple outcomes. A player in the game has to make a decision (action), and the outcome results from this action. The final result of all interactions between the players is the final outcome of the game result. More simply, we examine the beginning of a game (i_1) that two players start to achieve a possible outcome (a), and each player has only two possible actions to consider $\{A, R\}$, where A stands for *accept* and R stands for *reject*. Given that the set of actions for this outcome is $QoE(a) = \{A, R\}$, then the final outcome can be determined with the following formula:

$$QoE = QoE_{VD}(a) \times QoE_{TTP}(a) \times QoE_{SP}(a) \rightarrow \Omega_o \quad (6.12)$$

To present the game more clearly, we show different actions in *game i* for one possible outcome (a), as formulated in Eq. (6.13):

$$QoE(a) = QoE_{VD}(a) \times QoE_{TTP}(a) \quad (6.13)$$

Thus, four possible outcomes can occur for different combinations of players' actions, as in the following:

$$\begin{aligned} QoE_{VD|TTP}(A, A) &= a_1, QoE_{VD|TTP}(A, R) = a_2 \\ QoE_{VD|TTP}(R, A) &= a_3, QoE_{VD|TTP}(R, R) = a_4 \end{aligned} \quad (6.14)$$

The game can also be mapped onto the same outcome, as follows. Such an environment is unlikely in our system, since the outcome will remain the same regardless of the players' actions.

$$\begin{aligned} QoE_{VD|TTP}(A, A) &= a_1, QoE_{VD|TTP}(A, R) = a_1 \\ QoE_{VD|TTP}(R, A) &= a_1, QoE_{VD|TTP}(R, R) = a_1 \end{aligned} \quad (6.15)$$

We can also consider a more sensitive environment, where the outcomes could be affected by the action of one of the players as in the following:

$$\begin{aligned} QoE_{VD|TTP}(A, A) &= a_1, QoE_{VD|TTP}(A, R) = a_2 \\ QoE_{VD|TTP}(R, A) &= a_1, QoE_{VD|TTP}(R, R) = a_2 \end{aligned} \quad (6.16)$$

In this environment, it does not matter what the vehicle driver's action is, since the outcome depends only on the action of the trusted third party. If a trusted third party chooses to reject, as shown in Eq. (6.16), an a_2 outcome will result, while if the trusted third party chooses to accept an a_1 outcome will result. Our QoE game assumes that all players have influence in the game, thus all player selections affect the outcome. It becomes more interesting when we combine players' preferences with their actions. If we pick the most generic environment, where players' actions produce different outcomes as shown previously, and map it onto the players' preferences, we can predict the best possible outcomes of the game based on the players' selections, according to Example 1:

$$\begin{aligned}
QoE_{VD}(a_1) &= 2 & QoE_{VD}(a_2) &= 2 & QoE_{VD}(a_3) &= 1 & QoE_{VD}(a_4) &= 1 \\
QoE_{TTP}(a_1) &= 2 & QoE_{TTP}(a_2) &= 1 & QoE_{TTP}(a_3) &= 2 & QoE_{TTP}(a_4) &= 1
\end{aligned}$$

Since we know that each possible outcome is mapped onto a different action:

$$\begin{aligned}
QoE_{VD}(A, A) &= 2 & QoE_{VD}(A, R) &= 2 & QoE_{VD}(R, A) &= 1 & QoE_{VD}(R, R) &= 1 \\
QoE_{TTP}(A, A) &= 2 & QoE_{TTP}(A, R) &= 1 & QoE_{TTP}(R, A) &= 2 & QoE_{TTP}(R, R) &= 1
\end{aligned}$$

It is clear that the driver and the trusted third party action in this example means acceptance. The preference description is as the follows:

$$\begin{aligned}
QoE_{VD}(A, A) &\geq QoE_{VD}(A, R) \geq QoE_{VD}(R, A) \geq QoE_{VD}(R, R) \\
QoE_{TTP}(A, A) &\geq QoE_{TTP}(R, A) \geq QoE_{TTP}(A, R) \geq QoE_{TTP}(R, R)
\end{aligned}$$

Examining the different possible actions available to the driver and the trusted third party, raises the question: Which action should both chose to ensure best outcome? As explained earlier, the driver's best strategy to achieve the best possible outcome would be to *accept*. The trusted third party's best outcomes (i.e. a_1 and a_3) will be if vehicle driver *accepts* and they also accept, or if the driver rejects and they accept. However, the trusted third party preference description above means they accept over all other possible actions. Thus, in this example, it would be best if both players (VD and TTP) act rationally and chose the action that is mutually beneficial.

To prove the concept of using game models, we construct a different Example 2 with new player preferences as follows:

$$\begin{aligned}
QoE_{VD}(A, A) &= 2, & QoE_{VD}(A, R) &= 1, & QoE_{VD}(R, A) &= 2, & QoE_{VD}(R, R) &= 1, \\
QoE_{TTP}(A, A) &= 1, & QoE_{TTP}(A, R) &= 1, & QoE_{TTP}(R, A) &= 2, & QoE_{TTP}(R, R) &= 3,
\end{aligned}$$

In Example 2, the trusted third party's best outcome is to reject if the driver rejects, and the driver's best outcome is to accept and reject if the trusted third party always accepts.

However, the trusted third party's second best strategy in this game is to accept if the driver rejects. In this selection of (R, A) , both players get benefits and better outcomes.

6.5 PERFORMANCE EVALUATION

6.5.1 SIMULATION SETTINGS

The simulations for performance evaluation of the proposed schemes were performed using NS-2 [78]. The communication platform employs the Destination-Sequenced Distance-Vector Routing (DSDV) protocol, on IEEE 802.11.p communication stack with 1024 byte/packets. In the next set of simulations, each was repeated 10 times with a range of 5 to 50 vehicles. For the QoE-aware equation coefficient factors, the α, β and γ parameters are initially evaluated when equally weighted. In order to investigate the impact of the weighting on performance, we simulated random combinations against the game models, and these values can be adjusted based on the driver's preferences. With the simulations, we aim to: 1) study all possible parameters of the game model with different approaches, 2) study the impact of employing the concept of QoE-aware with gaming, 3) satisfy our proposed objectives for service provisioning in a vehicular cloud, 4) compare game theory-based and auction-based service provisioning approaches in vehicular clouds, and 5) evaluate the performance of game theoretic service provisioning approaches and determine the method that best fits the vehicular cloud environment. To illustrate this, we simulated multiple scenarios with a random number of game participants, in order to assess the behaviour of the proposed approaches in different situations.

To evaluate the impact of the QoE-aware coefficient factors (α, β , and γ) on the average savings for the MMIGS-WN model, we ran another set of simulations in which we diversified the weights of these factors (as shown in Table 6.1), and monitored the behaviour of the system. We studied the effect of the factors under these settings, while gradually increasing the network density from 10 to 50 vehicles in steps of five vehicles. In each step, the coefficient factor of the price (β) is increased equally by 20%, and α and γ are randomly set in intervals of $[0, 1 - \beta]$. We also introduced a new mode we call the *Driver's Comfort* mode, in which the MMIGS-WN game is played to provide

services that match the drivers' preferences. Thus, we evaluate the difference between user cost and privacy savings when the MMIGS-WN game is played with and without user preferences, denoting the former by *cost* and *privacy* and the latter by *cost-comfort*. The driver's comfort in this mode is represented by the driver's preferences at the beginning of each game. Table 6.1 presents these scenarios.

Table 6.1: Simulation scenarios and corresponding settings

<i>Scenarios</i>	<i># of</i>	<i>TTPs</i>	<i>SPs</i>	<i>a, β and γ in</i>	<i>a, β and γ in Driver's Comfort</i>
	<i>Vehicles</i>			<i>normal mode</i>	<i>mode enforcement</i>
1	5	3	4	0.33, 0.33, 0.34	—
2	10	4	7	0.33, 0.33, 0.34	$\beta = 20\%$, a and $\gamma = random$
3	15	6	8	0.33, 0.33, 0.34	$\beta = 40\%$, a and $\gamma = random$
4	20	9	10	0.33, 0.33, 0.34	$\beta = 60\%$, a and $\gamma = random$
5	25	10	14	0.33, 0.33, 0.34	$\beta = 80\%$, a and $\gamma = random$
6	50	12	21	0.33, 0.33, 0.34	$\beta = 100\%$, a and $\gamma = random$

6.5.2 SIMULATION RESULTS

We have evaluated the different proposed approaches in terms of the same metrics presented in section 0. We are also proposing a new metric namely, *MMIGS equilibrium*. MMIGS equilibrium is the average number of negotiation phases it takes the *MMIGS i(j)* approach to reach equilibrium among all participants. It is important to know the number of phases, as it is impractical for participants to engage in endless negotiations for a service.

We categorized the proposed models by two groups, namely the QoE unaware (i.e. *Game n*, *Game a* and *Neutral mode*) method and the QoE aware (i.e. MMIGS and auction-based) method, and conducted three types of evaluations and comparisons based on these methods. First, we analyzed how the proposed *Game n* and *Game a* models behave under the first three metrics (i.e. delay, price, and privacy), and compared them to the *Neutral mode* solution presented in [127]. *Neutral mode* is a simple model in which participants connect to the first available service provider without any QoE aspects. Secondly, another set of simulations was performed to explore the differences between the proposed game

approaches MMIGS with no negotiation (MMIGS-NN) and MMIGS with negotiation (MMIGS-WN) under the same metrics. These models were also compared to the auction-based model solution in [127]. The auction-based QoE service provisioning model uses QoE-aware, and manages a competition between trusted third parties and service providers to select auction participants that maximize their profit. Third, the last set of simulations investigated the impact of the MMIGS-WN in terms of average system savings and game equilibrium.

In the first group, we tested the average delay of the *Game n* and *Game a* models, and compared them to the *Neutral mode* with the delay randomly distributed, as shown in Figure 6.7. When the vehicles are sparse in the cloud network (i.e. 5 to 15 vehicles), *Game n* and *Game a* have delays similar to those of the neutral mode. As more vehicles enter the game the delay begins to increase, and is at its worst with a group of 50 vehicles. Thus, the delay under *Game a* is expected to be highest, since the worst two stages of communication are among its participants. However, as shown in Figure 6.8, latency in *Game a* improves the service cost.

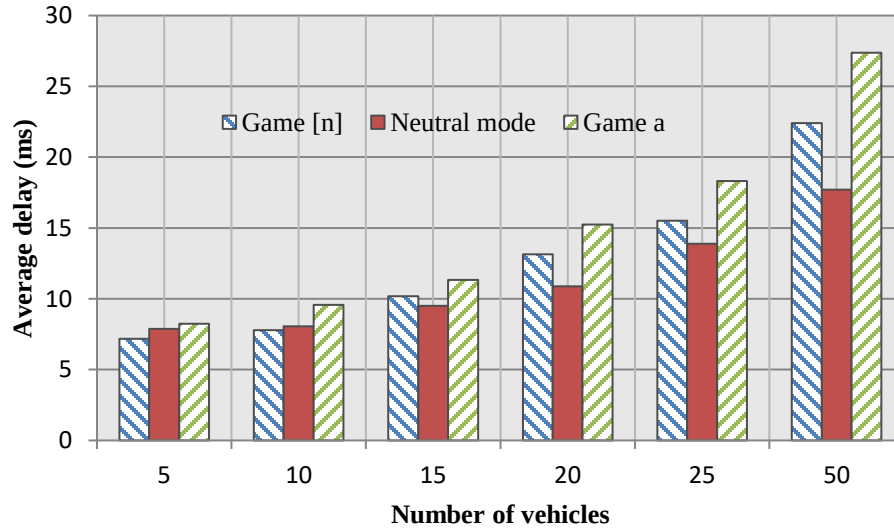


Figure 6.7: Game n delay

Figure 6.8 compares the average service cost of *Game n*, *Game a* and the *Neutral mode*, distributed over 50 vehicles. The *Neutral mode* technique engages the first available

nearby service provider to handle a driver service request. As service provider charges are unpredictable, a driver can find a service provider with lower charges without committing to them. However, *Game n* is a request to play a game while *Game a* is a blind game, which implies that a service provider who can make revenue might accept driver requests. A service provider should present a reasonable offer to the driver if they want to provide the service otherwise the driver has no reason to play the game. This is the main reason why the total service cost under *Game n* and *Game a* is less than that under the *Neutral mode*.

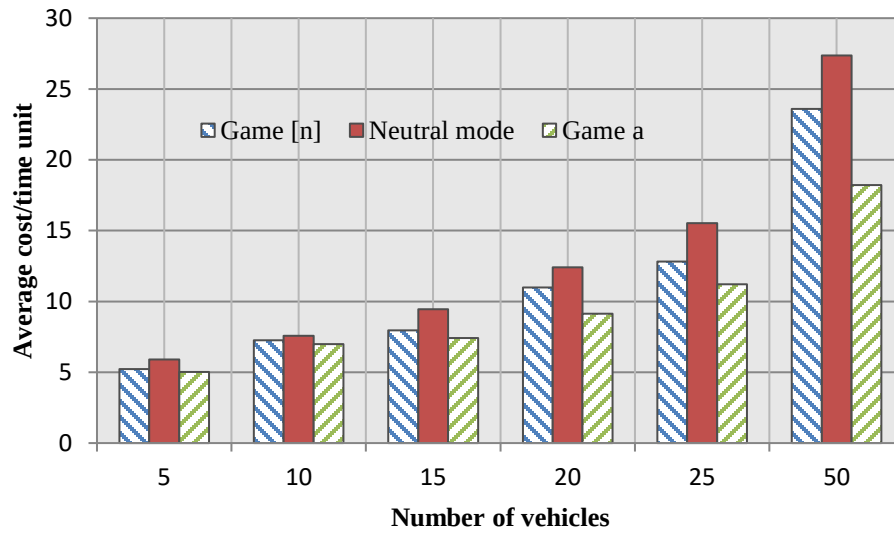


Figure 6.8: *Game n* service cost

The amount of released information with both modes is similar under a sparsely deployed vehicular network. As the vehicular network becomes denser the amount of revealed information increases, which makes the *Neutral mode* more beneficial than the game-based models, as shown in Figure 6.9. This is because the game models have more communication overhead than the *Neutral mode*, which has more information. Clearly, there is no reason for a driver to adopt the *Game a* or *Game n* model, since either could encumber them with additional delay and insignificant cost savings. Furthermore, there is no privacy guarantee due to the lack of QoE-aware in these models.

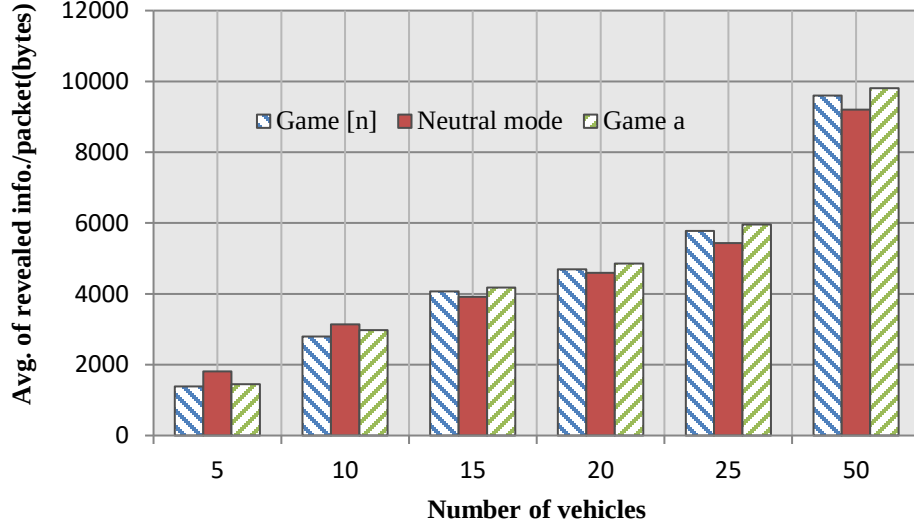


Figure 6.9: Game n privacy

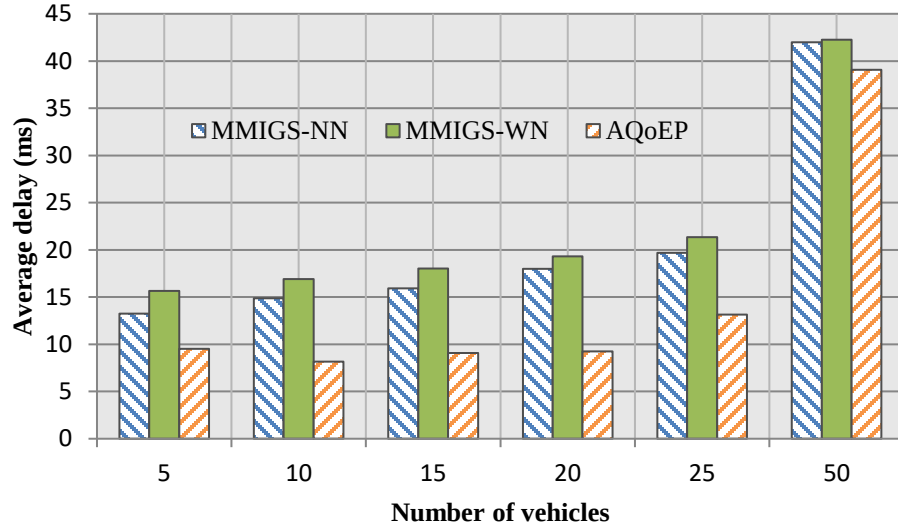


Figure 6.10: The impact of the QoE-aware models on service delay

Figure 6.10 illustrates the total impact of QoE-aware on average delay in every group of vehicles when it is integrated with different models. We observed that, under auction-based provisioning (i.e. AQoEP), there are lower average delays until the vehicular network has 50 vehicles. Due to the number of communications levels, both MMIGS-NN and MMIGS-WN lead to longer delays than AQoEP. MMIGS-NN has two levels of communication (i.e. i_1 and i_2), while MMIGS-WN has a minimum of two levels of

communication (i.e. i_1 , i_2 , j_1 and j_2). Once the network has 50 vehicles, MMIGS-NN and MMIGS-WN have almost the same delay, and they reduce the delay gap between them and AQoEP from 6.13ms to 3.20ms (48.6%). This can be explained by the fact that MMIGS-WN appears to develop its experience over time, and compromises with a little delay in each group participant experience ($\sim 3\%$). Increasing the average delay also means more negotiations, which indicates that the participants are engaged in the game. Moreover, AQoEP conducts one level of communications while MMIGS conducts at least two levels; this can be considered a significant improvement.

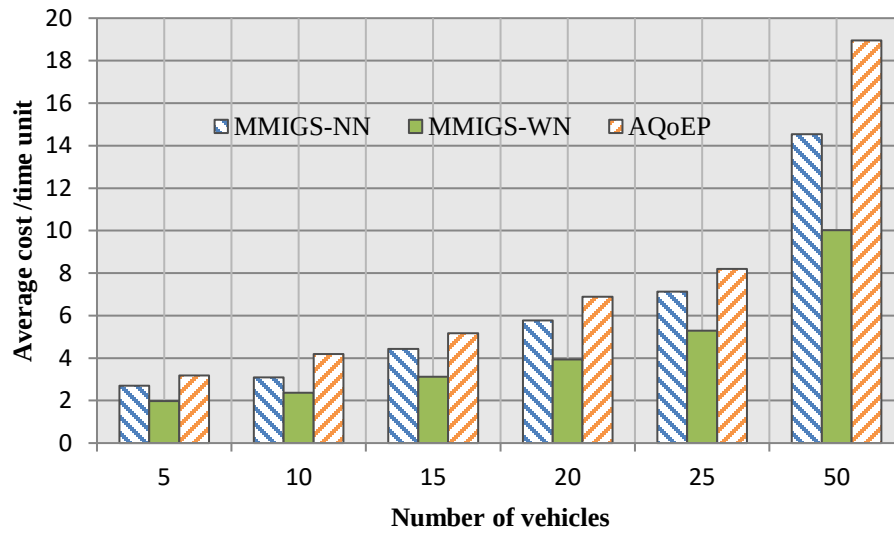


Figure 6.11: The impact of QoE-aware models on service cost

As shown in Figure 6.11, MMIGS-WN outperforms both the other models with respect to service cost. The main reason for this is the circumstances of the negotiations among the game participants, which is not relevant in the case of AQoEP. The average service cost difference between the proposed MMIGS-WN and AQoEP is in the range of 50% in the first five scenarios, and approximately 65% in the last scenario. MMIGS-WN is also less costly than MMIGS-NN; up to 15% in a sparse vehicular network, and approximately 23% in a denser network. This means that the proposed game models achieve savings on the provision of services of up to 50%. Figure 6.12 shows the impact on the amount of revealed information when QoE-aware is adopted with the game theory concept. MMIGS-WN again outperforms the other two models. AQoEP seems to be vulnerable

with respect to driver privacy, so MMIGS-WN is a promising alternative when drivers are concerned about revealing their information. It improves driver privacy up to 47% compared to AQoEP, and 19% compared to MMIGS-NN. These improvements in service cost and driver privacy are due to the following: 1) using game theory approaches among vehicular participants, 2) adopting QoE-aware with games, and 3) testing the feasibility of integrating a negotiation stage between game participants.

The service cost savings with or without applying the *Driver's Comfort* mode in the MMIGS-WN game are illustrated in Figure 6.13, where the average savings under the MMIGS-WN model are compared to when the MMIGS-WN game is played with prior knowledge of the driver's comfort preferences. As the weight of β increases, the average savings increase and the privacy savings decrease. With the *Driver's Comfort* mode savings, there is always a gap in the 20% to 40% range and the 80% to 100% range. The best average cost/privacy savings are achieved with service weight costs of 60%. Under this weight setting, the average difference in savings between the MMIGS-WN model and the MMIGS-WN model with *Driver's Comfort Mode* is less than 5% of the total *Driver's Comfort* savings.

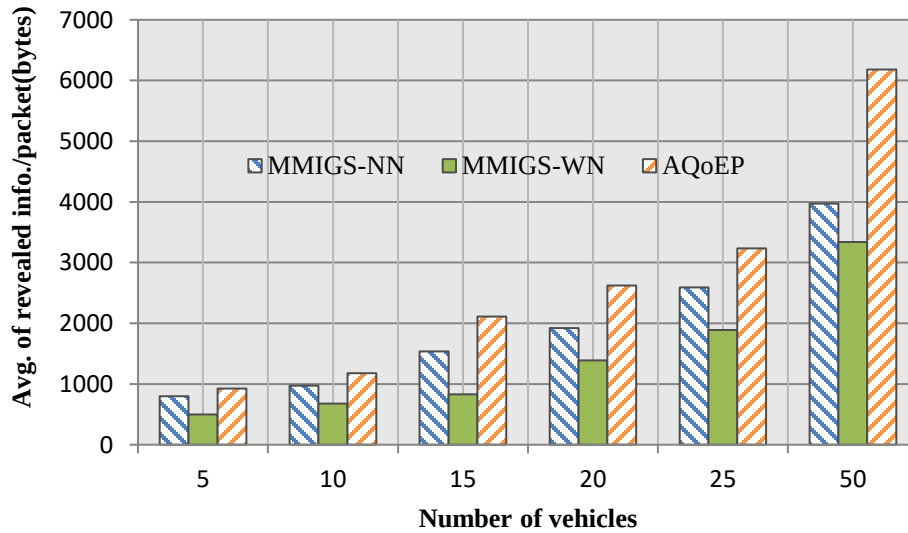


Figure 6.12: The impact of the QoE-aware models on privacy

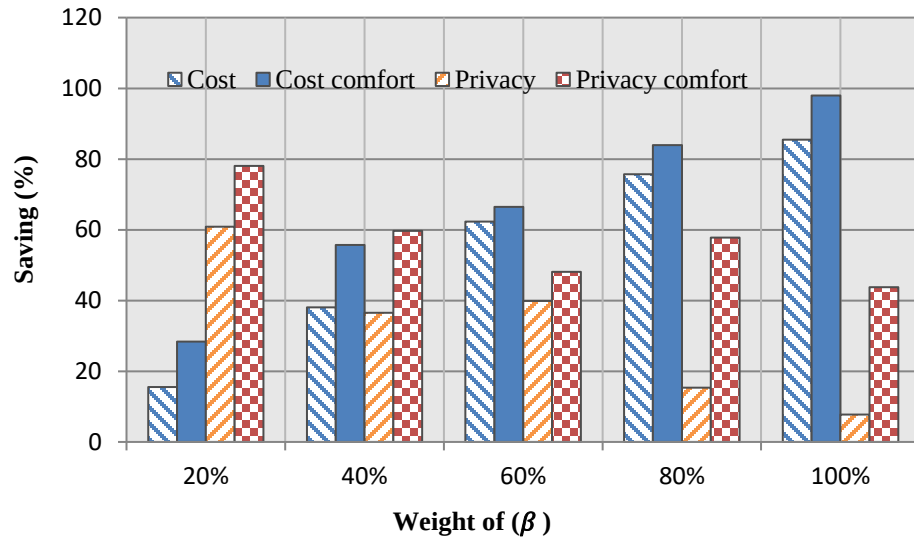


Figure 6.13: MMIGS-WN average percentage saving

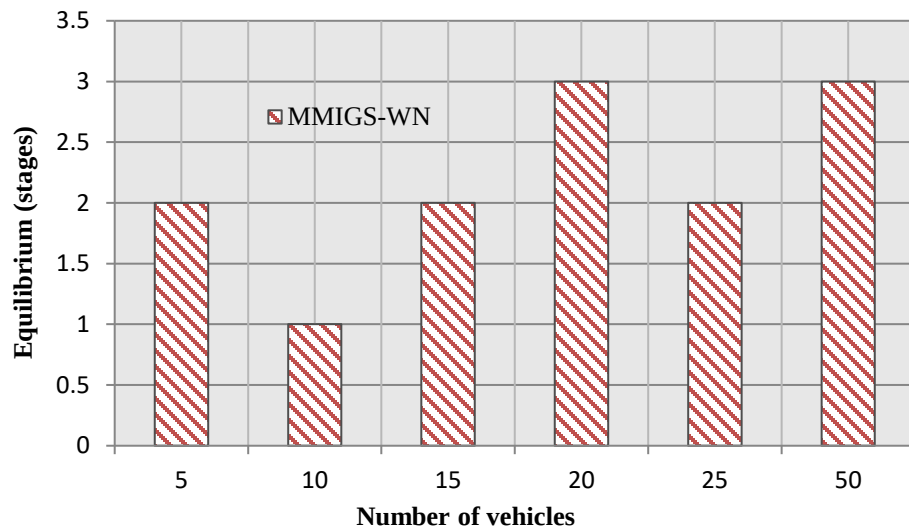


Figure 6.14: MMIGS-WN equilibrium

In the final simulation set, we tested the average number of equilibrium stages required for the MMIGS-WN model to become stable. We seek answers for two questions, namely ‘How many stages are required for a game to reach equilibrium in the proposed scenarios?’ and ‘Does the number of participants have an impact on the number of stages?’ Figure 6.14 illustrates the equilibrium stages against the number of vehicles, and shows that the number of the vehicles does not affect the number of equilibrium stages,

since the system can have the same number of stages (i.e. 3) with 50 vehicles or 20 vehicles.

6.6 SUMMARY

In this chapter, we proposed a multiagent/multiobjective interaction game system (MMIGS) for service provisioning in a vehicular cloud, based on a game theoretic approach and a Quality of Experience (QoE) framework. MMIGS balances the overall game, while enhancing drivers' service costs and preserving their privacy. The proposed game system differs from other conventional models, as it allows drivers to prioritize their preferences. It also takes QoE-aware into account, which enables drivers to negotiate the terms of their preferences. Our extensive simulations show that the proposed game model with negotiations (i.e. MMIGS-WN), incorporated with QoE awareness and a trusted third party (TTP), efficiently mitigates communication latency by a bounded percentage of 3%. In addition, MMIGS-WN with QoE-aware and TTP involvement achieves reduced service costs of 65%, and preserves driver privacy (i.e. information revealed) by 47%. We also analyzed the performance of this game model in scenarios where the number of vehicles and the weighting factors varied, and the results showed that significant savings can be achieved under various weighted combinations of driver preferences. Finally, we calculated the total number of stages required for the game to reach equilibrium, and determined that the number of vehicles does not affect the number of equilibrium stages.

CHAPTER 7 CONCLUSIONS AND FUTURE RESEARCH

7.1 CONCLUDING REMARKS

In this thesis, the issues of service provision in a vehicular cloud are investigated in detail, with a focus on latency, cost and privacy. Since service provision for vehicle drivers is a challenging task, it is attracting growing attention from the research community. Drivers have identified many concerns related to engaging with vehicular environments, particularly, as mentioned, service latency, service cost, and the degree of information revealed to service providers (i.e. privacy). Though several studies have been proposed to resolve some of these issues, there is currently no means of ensuring that drivers get the best price, lowest latency and highest privacy. Thus, the main focus of this work is to propose and develop a novel framework to address these issues in a holistic manner.

We introduce the concept of using a Vehicular Trusted Third Party (VTTP) in vehicular clouds to manage provisioning services between vehicle drivers and service providers. We demonstrate that using VTTP is a viable approach, as it is already used for several applications. The proposed scheme was analyzed in a variety of scenarios, including light and heavy services. Through numerical analysis, we show how it leads to promising results compared to conventional models. We know that network density and the type of service play major roles in service latency. Our approach addresses and enhances the latency issue.

We highlighted the significance of Quality of Experience (QoE) in vehicular cloud systems, and reviewed the most relevant related work. We then proposed a framework for vehicular provisioning services, namely Vehicular Provisioning Services Quality of Experience (VPSQoE), which utilizes the concept of TTP between drivers and service providers. The objectives of the framework are: 1) to provide several provisioning services in vehicular clouds at low cost, 2) to do so with minimal revealed information, and 3) with minimum service latency. We formulated the QoE based on vehicular users' degree of satisfaction, through a rating/recommendation system. The numerical results

show that QoE-based provisioning services in vehicular clouds can be a compromise between naïve provisioning services, and other provisioning approaches that target a specific user requirement (i.e. cost, latency or revealed information).

We propose a novel approach based on auction theory and QoE in vehicular clouds, namely Auction-driven Quality of Experience Provisioning (AQoEP). The procedure to select VPSQoE framework users (i.e. drivers, trusted third parties and service providers) is done automatically by auction, in order to maximize participants' revenue and best address their interests and requirements. AQoEP has been compared to the other approaches that are commonly used to measure the service latency, cost and driver privacy in vehicular clouds. We found that the auction-based approach provides better results in terms of service latency and the amount of revealed information with significant service cost reductions.

Finally, we propose an interaction game system for service provision in vehicular clouds, Multiagent/Multiobjective Interaction Game System for Service Provisioning in Vehicular Clouds (MMIGS). The model is based on game theory and the QoE approach developed in Chapters 3 and 4. Extensive simulations showed that the proposed game model, With Negotiations (i.e. MMIGS-WN), incorporated with QoE awareness and TTP, efficiently mitigates communication latency by a bounded percentage of 3%, reduces service costs up to 65%, and preserves driver privacy by 47%.

7.2 FUTURE RESEARCH

Based on the investigations and conclusions in this thesis, there are several areas that could be considered for future research. These are summarized as follows:

- Extend the proposed MMIGS model for use with mobile vehicles, as this model assumed a network of fixed/parked vehicles. Policy-based location-aware mobility is considered to be adaptable to MMIGS.
- Incorporate “sensing as a service” with the vehicular cloud, and introduce trustworthiness into the proposed framework as suggested in the related work [134].

- Apply additional metrics to quantify driver privacy, and integrate other methods with the proposed framework to hide sensitive data, such as drivers' mobility patterns. Prevent service providers from learning driver identities by cloaking regions to avoid revealing location information.
- Develop an extension of the proposed MMIGS model to define automated game selection between game participants, and design a suitable optimization scheme that effectively addresses this problem.
- Incorporate different types of services and a recent mobility model with the auction approach, to study its impact on the level of complexity of the auction. These two factors have not yet been considered, and we anticipate they will have a positive impact on the auction if we use a suitable mobility model with different type of services.
- Incorporate the techniques of prefetching and caching to reduce communication latency. Many researchers have advocated the use of prefetching and caching to reduce communication latency. Several performance improvements are expected from adopting such techniques.

REFERENCES

- [1] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless Communications and Mobile Computing*, Wiley Online Library, vol. 13, no. 18, pp. 1587–1611, 2013.
- [2] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," *Communications of the ACM*, ACM, vol. 53, no. 4, pp. 50–58, 2010.
- [3] M. A. Rappa, "The Utility Business Model and the Future of Computing Services," *IBM Systems Journal*, IBM Corporation, vol. 43, no. 1, pp. 32–42, 2004.
- [4] M. R. Rahimi, J. Ren, C. H. Liu, A. V. Vasilakos, and N. Venkatasubramanian, "Mobile Cloud Computing: A Survey, State of Art and Future Directions," *Mobile Networks and Applications*, Springer, vol. 19, no. 2, pp. 133–143, 2014.
- [5] S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic, "The Next Paradigm Shift: From Vehicular Networks to Vehicular Clouds," *Mobile Ad Hoc Networking: The Cutting Edge Directions*, Wiley Online Library, pp. 645–700, 2013.
- [6] M. Gerla, "Vehicular Cloud Computing," in *Proceeding of the 2012 Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, 2012, pp. 152–155.
- [7] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin, and T. Weil, "Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions," *IEEE Communications Surveys and Tutorials*, vol. 13, no. 4, pp. 584–616, 2011.
- [8] S. Abdelhamid, H. S. Hassanein, and G. Takahara, "Vehicle as a resource (VaaR)," *IEEE Network*, vol. 29, no. 1, pp. 12–17, 2015.
- [9] M. Whaiduzzaman, M. Sookhak, A. Gani, and R. Buyya, "A Survey on Vehicular Cloud Computing," *Journal of Network and Computer Applications*, Academic Press Ltd., vol. 40, pp. 325–344, 2014.
- [10] M. Aloqaily, B. Kantarci, and H. T. Mouftah, "Vehicular clouds: State of the art, Challenges and Future Directions," in *Proceedings of the 2015 IEEE Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, 2015, pp. 1–6.
- [11] M. A. Razzaque, A. Salehi, and S. M. Cheraghi, "Wireless Networks and Security: Issues, Challenges and Research Trends," *Signals and Communication Technology*, Springer, vol. 1, no. 1, pp. 107–132, 2013.
- [12] M. N. Mejri, J. Ben-Othman, and M. Hamdi, "Survey on {VANET} security challenges and possible cryptographic solutions," *Journal of Vehicular Communications*, Elsevier, vol. 1, no. 2, pp. 53–66, 2014.
- [13] S. Pearson and A. Benameur, "Privacy, Security and Trust Issues Arising from Cloud Computing," in *Proceedings of the 2010 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, 2010, pp. 693–702.

- [14] R. S. Schwartz, R. R. R. Barbosa, N. Meratnia, G. Heijenk, and H. Scholten, "A directional data dissemination protocol for vehicular environments," *Journal of Computer Communications, Elsevier*, vol. 34, no. 17, pp. 2057–2071, 2011.
- [15] J. Zhao, C. Wu, and Z. Li, "Cost Minimization in Multiple IaaS Clouds: A Double Auction Approach," *arXiv preprint arXiv:1308.0841*, vol. abs/1308.0841, 2013.
- [16] W. Chen, R. K. Guha, T. J. Kwon, J. Lee, and Y.-Y. Hsu, "A survey and challenges in routing and data dissemination in vehicular ad hoc networks," *Journal of Wireless Communications and Mobile Computing, John Wiley & Sons, Ltd.*, vol. 11, no. 7, pp. 787–795, 2011.
- [17] H. S. Hassanein, S. Abdelhamid, and K. Elgazzar, "A framework for vehicular cloud computing," in *Proceedings of the 2015 International Conference on Connected Vehicles and Expo (ICCVE)*, 2015, pp. 238–239.
- [18] G. Yan, D. Wen, S. Olariu, and M. C. Weigle, "Security challenges in vehicular cloud computing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 284–294, 2013.
- [19] F. Kargl, P. Papadimitratos, L. Buttyan, M. Mter, E. Schoch, B. Wiedersheim, T. V. Thong, G. Calandriello, A. Held, A. Kung, and J. P. Hubaux, "Secure vehicular communication systems: implementation, performance, and research challenges," *IEEE Communications Magazine*, vol. 46, no. 11, pp. 110–118, 2008.
- [20] K. Plohl and H. Federrath, "A privacy aware and efficient security infrastructure for vehicular ad hoc networks," *Journal of Computer Standards and Interfaces: State of standards in the information systems security area, Elsevier*, vol. 30, no. 6, pp. 390–397, 2008.
- [21] S. Ren and M. van der Schaar, "Dynamic Scheduling and Pricing in Wireless Cloud Computing," *IEEE Transactions on Mobile Computing*, vol. 13, no. 10, pp. 2283–2292, 2014.
- [22] J. Luo, L. Rao, and X. Liu, "Temporal Load Balancing with Service Delay Guarantees for Data Center Energy Cost Optimization," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 3, pp. 775–784, 2014.
- [23] R. Ranchal, B. Bhargava, L. B. Othmane, L. Lilien, A. Kim, M. Kang, and M. Linderman, "Protection of Identity Information in Cloud Computing without Trusted Third Party," in *Proceedings of the 2010 IEEE Symposium on Reliable Distributed Systems (RDS)*, 2010, pp. 368–372.
- [24] G. Yan, D. B. Rawat, and B. B. Bista, "Towards Secure Vehicular Clouds," in *Proceedings of the 2012 International Conference on Complex Intelligent and Software Intensive Systems (CISIS)*, 2012, pp. 370–375.
- [25] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: state-of-the-art and research challenges," *Journal of Internet Services and Applications, Springer open*, vol. 1, no. 1, pp. 7–18, 2010.
- [26] C. I. report, *Global Smartphone Market Analysis and Outlook: Disruption in a Changing Market*. Global Smartphone Market Analysis, 2014.
- [27] H. T. Mouftah and B. Kantarci, *Communication Infrastructures for Cloud Computing: Design and Applications*. IGI Global, Hershey, Pennsylvania, USA, ISBN13: 9781466645226, 2013.
- [28] W. H. Organization, "Supporting a decade of action," *World Health Organization (WHO) - global status report on road safety 2013*. 2013.

- [29] R. Hussain, J. Son, H. Eun, S. Kim, and H. Oh, "Rethinking vehicular communications: Merging VANET with cloud computing," in *Proceedings of the 2012 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, 2012, pp. 606–609.
- [30] S. Olariu, M. Eltoweissy, and M. Younis, "Towards autonomous vehicular clouds," *Transactions on Mobile Communications and Applications*, Springer, vol. 11, no. 1, 2011.
- [31] S. Olariu, T. Hristov, and G. Yan, "The next paradigm shift: from vehicular networks to vehicular clouds," *Mobile Ad Hoc Networking*, Wiley Online Library, vol. 1, no. 1, pp. 645–700, 2013.
- [32] M. Gerla, E.-K. Lee, G. Pau, and U. Lee, "Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds," in *Proceedings of the 2014 IEEE World Forum on Internet of Things (WF-IoT)*, 2014, pp. 241–246.
- [33] D. Wu, Y. Zhang, L. Bao, and A. C. Regan, "Location-Based Crowdsourcing for Vehicular Communication in Hybrid Networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 837–846, 2013.
- [34] X. Hu, V. C. M. Leung, K. G. Li, E. Kong, H. Zhang, N. S. Surendrakumar, and P. TalebiFard, "Social Drive: A Crowdsourcing-based Vehicular Social Networking System for Green Transportation," in *Proceedings of the 2013 ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications (DIVANet)*, 2013, pp. 85–92.
- [35] M. Faezipour, M. Nourani, A. Saeed, and S. Addepalli, "Progress and Challenges in Intelligent Vehicle Area Networks," *ACM Communications Magazine*, vol. 55, no. 2, pp. 90–100, 2012.
- [36] R. I. Meneguet, "A Vehicular Cloud-Based Framework for the Intelligent Transport Management of Big Cities," *International Journal of Distributed Sensor Networks*, Hindawi Publishing Corporation, vol. 12, no. 5, pp. 1–9, 2016.
- [37] D. Jiang and L. Delgrossi, "IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments," in *Proceedings of the 2008 IEEE Vehicular Technology Conference (VTC)*, 2008, pp. 2036–2040.
- [38] Q. Xu, T. Mak, J. Ko, and R. Sengupta, "Layer-2 protocol design for vehicle safety communications in dedicated short range communications spectrum," in *Proceedings of the 2004 IEEE International Conference on Intelligent Transportation Systems (ITS)*, 2004, pp. 1092–1097.
- [39] R. Hussain, S. Kim, and H. Oh, "Towards Privacy Aware Pseudonymless Strategy for Avoiding Profile Generation in VANET," *Information Security Applications*, Springer, vol. 5932, pp. 268–280, 2009.
- [40] C. Rong, S. T. Nguyen, and M. G. Jaatun, "Beyond lightning: A survey on security challenges in cloud computing," *Computers and Electrical Engineering: on Recent Advanced Technologies and Theories for Grid and Cloud Computing and Bio-engineering*, Elsevier, vol. 39, no. 1, pp. 47–54, 2013.
- [41] H. Suo, Z. Liu, J. Wan, and K. Zhou, "Security and privacy in mobile cloud computing," in *Proceedings of the 2013 International Conference on Wireless Communications and Mobile Computing (IWCMC)*, 2013, pp. 655–659.

- [42] J. Yang and Z. Chen, "Cloud computing research and security issues," in *Proceedings of the 2010 International conference on computational intelligence and software engineering (CiSE)*, 2010, pp. 1–3.
- [43] M. Mun, S. Hao, N. Mishra, K. Shilton, J. Burke, D. Estrin, M. Hansen, and R. Govindan, "Personal Data Vaults: A Locus of Control for Personal Data Streams," in *Proceedings of the 2010 International Conference on emerging Networking EXperiments and Technologies (CoNEXT)*, 2010, pp. 1–12.
- [44] E. Fonseca, A. Festag, R. Baldessari, and R. L. Aguiar, "Support of anonymity in vanets-putting pseudonymity into practice," in *Proceedings of the 2007 IEEE Wireless Communications and Networking Conference (WCNC)*, 2007, pp. 3400–3405.
- [45] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *Proceedings of the 2004 Eurocrypt - Lecture Notes in Computer Science (LNCS)*, 2004, pp. 506–522.
- [46] K. Ren, W. Lou, K. Kim, and R. Deng, "A novel privacy preserving authentication and access control scheme for pervasive computing environments," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 4, pp. 1373–1384, 2006.
- [47] U. Habiba, R. Masood, M. A. Shibli, and M. A. Niazi, "Cloud identity management security issues & solutions: a taxonomy," *Complex Adaptive Systems Modeling, Springer Open*, vol. 2, no. 1, pp. 1–37, 2014.
- [48] K. Anastasopoulou, T. Tryfonas, and S. Kokolakis, "Strategic Interaction Analysis of Privacy-Sensitive End-Users of Cloud-Based Mobile Apps," in *Proceedings of the 2013 International Conference on Human Aspects of Information Security, Privacy, and Trust (HAS)*, 2013, pp. 209–216.
- [49] L. Rajbhandari and E. A. Sneekenes, "Using Game Theory to Analyze Risk to Privacy: An Initial Insight," *Privacy and Identity Management for Life, IFIP Advances in Information and Communication Technology, Springer*, vol. 352, no. 1, pp. 41–51, 2011.
- [50] K. Hausken, "Probabilistic risk analysis and game theory," *Risk Analysis, Wiley Online Library*, vol. 22, no. 1, pp. 17–27, 2002.
- [51] R. Campbell, J. Al-Muhtadi, P. Naldurg, G. Sampemane, and M. D. Mickunas, "Towards security and privacy for pervasive computing," in *Proceedings of the 2002 Next-NSF-JSPS international conference on Software security: theories and systems (ISSS)*, 2003, pp. 1–15.
- [52] B. Parno and A. Perrig, "Challenges in securing vehicular networks," in *Proceedings of the 2005 ACM Workshop on hot topics in networks (HotNets-IV)*, 2005, pp. 1–6.
- [53] S. Olariu, I. Khalil, and M. Abuelela, "Taking VANET to the clouds," *International Journal of Pervasive Computing and Communications, Emerald Group Publishing Ltd*, vol. 7, no. 1, pp. 7–21, 2011.
- [54] M. Raya and J.-P. Hubaux, "Securing Vehicular Ad Hoc Networks," *Journal of Computer Security: on Security of Ad-hoc and Sensor Networks, IOS Press*, vol. 15, no. 1, pp. 39–68, 2007.
- [55] M. Raya and J.-P. Hubaux, "The Security of Vehicular Ad Hoc Networks," in *Proceedings of the 2005 ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN)*, 2005, pp. 11–21.

- [56] M. Raya, P. Papadimitratos, and J.-P. Hubaux, "Securing Vehicular Communicaions," *IEEE Wireless Communications*, vol. 13, no. 5, pp. 8–15, 2006.
- [57] P. Kamat, A. Baliga, and W. Trappe, "An Identity-based Security Framework For VANETs," in *Proceedings of the 2006 ACM International Workshop on Vehicular Ad Hoc Networks (VANET)*, 2006, pp. 94–95.
- [58] J. Sun, C. Zhang, Y. Zhang, and Y. Fang, "An identity-based security system for user privacy in vehicular ad hoc networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 9, pp. 1227–1239, 2010.
- [59] K. Ibrahim and M. C. Weigle, "CASCADE: Cluster-based accurate syntactic compression of aggregated data in VANETs," in *Proceedings of the 2008 IEEE Globecom Workshops on Automotive Networking and Applications (Autonet)*, 2008, pp. 1–10.
- [60] S. Dietzel, J. Gürtler, and F. Kargl, "A resilient in-network aggregation mechanism for VANETs based on dissemination redundancy," *Ad Hoc Networks, Elsevier*, vol. 37, pp. 101–109, 2016.
- [61] P. Golle, K. Leyton-Brown, I. Mironov, and M. Lillibridge, "Incentives for Sharing in Peer-to-Peer Networks," in *Proceedings of the 2001 ACM conference on Electronic Commerce (CEC)*, 2001, pp. 75–87.
- [62] M. Salek, S. Shayandeh, and D. Kempe, "You Share, I Share: Network Effects and Economic Incentives in P2P File-Sharing Systems," in *Proceedings of the 2011 International Conference on Internet and Network Economics (WINE)*, 2011, pp. 354–365.
- [63] P. Michiardi and R. Molva, "Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *Proceedings of the 2012 Advanced communications and multimedia security (ACMS)*, 2002, pp. 107–121.
- [64] S. Zhong, J. Chen, and Y. R. Yang, "Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks," in *Proceedings of the 2003 Joint Conference of the IEEE Computer and Communications (ICC) and INFOCOM*, 2003, pp. 1987–1997.
- [65] P. Caballero-Gil, J. Molina-Gil, C. Hernández-Goya, and C. Caballero-Gil, "Stimulating cooperation in self-organized vehicular networks," in *Proceedings of the 2009 Asia-Pacific Conference on Communications (ACC)*, 2009, pp. 346–349.
- [66] T. Chen, L. Wu, F. Wu, and S. Zhong, "Stimulating cooperation in vehicular ad hoc networks: a coalitional game theoretic approach," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 2, pp. 566–579, 2011.
- [67] Y.-W. Lin, Y.-S. Chen, and S.-L. Lee, "Routing Protocols in Vehicular Ad Hoc Networks: A Survey and Future Perspectives," *Journal of Information Science and Engineering, JISE*, vol. 26, no. 3, pp. 913–932, 2010.
- [68] Y.-S. Chen, C.-S. Hsu, and Y.-T. Jiang, "A delay-bounded routing protocol for vehicular ad hoc networks with traffic lights," *Wireless Communications and Mobile Computing, John Wiley & Sons, Ltd*, vol. 15, no. 11, pp. 1577–1588, 2015.
- [69] C. E. Palazzi, F. Pezzoni, and P. M. Ruiz, "Delay-bounded data gathering in urban vehicular sensor networks," *Pervasive and Mobile Computing, Elsevier*, vol. 8, no. 2, pp. 180–193, 2012.
- [70] I. Menache, A. Ozdaglar, and N. Shimkin, "Socially Optimal Pricing of Cloud Computing Resources," in *Proceedings of the 2011 International Conference on*

- Performance Evaluation Methodologies and Tools (VALUETOOLS)*, 2011, pp. 322–331.
- [71] Q. Wang, K. Ren, and X. Meng, “When cloud meets eBay: Towards effective pricing for cloud computing,” in *Proceedings of the 2012 IEEE International Conference on Computer Communications (INFOCOM)*, 2012, pp. 936–944.
 - [72] L. Rao, X. Liu, L. Xie, and W. Liu, “Minimizing Electricity Cost: Optimization of Distributed Internet Data Centers in a Multi-Electricity-Market Environment,” in *Proceedings of the 2010 IEEE International Conference on Computer Communications (INFOCOM)*, 2010, pp. 1–9.
 - [73] M. U. Banu and K. Saravanan, “Optimizing the Cost for Resource Subscription Policy in IaaS Cloud,” *arXiv preprint arXiv:1402.2491*, 2014.
 - [74] M. Mihailescu and Y. M. Teo, “Dynamic Resource Pricing on Federated Clouds,” in *Proceedings of the 2010 IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid)*, 2010, pp. 513–517.
 - [75] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of Things (IoT): A vision, architectural elements, and future directions,” *Future Generation Computer Systems, Elsevier*, vol. 29, no. 7, pp. 1645–1660, 2013.
 - [76] E. Lee, L. EunKy, M. Gerla, and S. Y. Oh, “Vehicular cloud networking: architecture and design principles,” *IEEE Communications Magazine*, vol. 52, no. 2, pp. 148–155, 2014.
 - [77] M. Aloqaily, B. Kantarci, and H. T. Mouftah, “Provisioning delay effect of partaking a Trusted Third Party in a vehicular cloud,” in *Proceedings of the 2014 Global Information Infrastructure and Networking Symposium (GIIS)*, 2014, pp. 1–3.
 - [78] K. Fall and E. Kannan, “The ns Manual (formerly ns Notes and Documentation).” 2011.
 - [79] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, “A comprehensive survey on vehicular Ad Hoc network,” *Journal of network and computer applications, Elsevier*, vol. 37, pp. 380–392, Jan. 2014.
 - [80] T. K. Refaat, B. Kantarci, and H. T. Mouftah, “Virtual machine migration and management for vehicular clouds,” *Vehicular Communications, Elsevier*, vol. 4, pp. 47–56, Apr. 2016.
 - [81] H. R. Arkian, R. E. Atani, A. Diyanat, and A. Pourkhalili, “A cluster-based vehicular cloud architecture with learning-based resource management,” *The Journal of Supercomputing, Springer*, vol. 71, no. 4, pp. 1401–1426, 2015.
 - [82] F. Dressler, P. Handle, and C. Sommer, “Towards a Vehicular Cloud - Using Parked Vehicles As a Temporary Network and Storage Infrastructure,” in *Proceedings of the 2014 ACM International Workshop on Wireless and Mobile Technologies for Smart Cities (WiMobCity)*, 2014, pp. 11–18.
 - [83] M. Siller and J. Woods, “Improving quality of experience for multimedia services by QoS arbitration on a QoE framework,” in *Proceedings of the 2003 Packed Video Workshop*, 2003, pp. 1–7.
 - [84] K. Kilkki, “Quality of experience in communications ecosystem,” *Journal of Universal Computer Science*, vol. 14, no. 5, pp. 615–624, 2008.

- [85] A. Hamam and A. El Saddik, "Toward a mathematical model for quality of experience evaluation of haptic applications," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 12, pp. 3315–3322, 2013.
- [86] A. Hamam, M. Eid, and A. El Saddik, "Effect of kinesthetic and tactile haptic feedback on the quality of experience of edutainment applications," *Multimedia tools and applications*, Springer, vol. 67, no. 2, pp. 455–472, 2013.
- [87] M. Jarschel, D. Schlosser, S. Scheuring, and T. Hobfeld, "An evaluation of QoE in cloud gaming based on subjective tests," in *Proceedings of the 2011 International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*, 2011, pp. 330–335.
- [88] K. U. R. Laghari and K. Connelly, "Toward total quality of experience: A QoE model in a communication ecosystem," *IEEE Communications Magazine*, vol. 50, no. 4, pp. 58–65, 2012.
- [89] R. K. Mok, E. W. Chan, X. Luo, and R. K. Chang, "Inferring the QoE of HTTP video streaming from user-viewing activities," in *Proceedings of the 2011 ACM SIGCOMM workshop on Measurements up the stack (W-MUST)*, 2011, pp. 31–36.
- [90] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hobfeld, and P. Tran-Gia, "A Survey on Quality of Experience of HTTP Adaptive Streaming," *IEEE Communications Surveys and Tutorials*, vol. 17, no. 1, pp. 469–492, 2015.
- [91] D. Rodriguez, Z. Wang, R. Rosa, and G. Bressan, "The impact of video-quality-level switching on user quality of experience in dynamic adaptive streaming over HTTP," *EURASIP Journal on Wireless Communications and Networking*, Springer Open, vol. 2014, no. 1, pp. 1–15, 2014.
- [92] M. Zhao, X. Gong, J. Liang, W. Wang, X. Que, and S. Cheng, "QoE-driven cross-layer optimization for wireless dynamic adaptive streaming of scalable videos over HTTP," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 3, pp. 451–465, 2015.
- [93] V. Menkovski, A. Oredope, A. Liotta, and A. C. Sánchez, "Predicting Quality of Experience in Multimedia Streaming," in *Proceedings of the 2009 ACM International Conference on Advances in Mobile Computing and Multimedia (MoMM)*, 2009, pp. 52–59.
- [94] W. Wu, A. Arefin, R. Rivas, K. Nahrstedt, R. Sheppard, and Z. Yang, "Quality of Experience in Distributed Interactive Multimedia Environments: Toward a Theoretical Framework," in *Proceedings of the 2009 ACM International Conference on Multimedia (MM)*, 2009, pp. 481–490.
- [95] A. Hamam, M. Eid, A. El Saddik, and N. D. Georganas, "A Quality of Experience Model for Haptic User Interfaces," in *Proceedings of the 2008 Ambi-Sys Workshop on Haptic User Interfaces in Ambient Media Systems (HAS)*, 2008, pp. 1–6.
- [96] A. Hamam and A. E. Saddik, "Toward a Mathematical Model for Quality of Experience Evaluation of Haptic Applications," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 12, pp. 3315–3322, Dec. 2013.
- [97] A. Hamam, M. Eid, A. El Saddik, and N. D. Georganas, "A Fuzzy Logic System for Evaluating Quality of Experience of Haptic-Based Applications," in *Proceedings of the 2008 International Conference on Haptics: Perception, Devices and Scenarios (HPDS)*, 2008, pp. 129–138.

- [98] A. Hamam, A. E. Saddik, and J. Alja'am, "A Quality of Experience Model for Haptic Virtual Environments," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 10, no. 3, pp. 1–23, 2014.
- [99] E. Dias, "A model to measure Quality of Experience (QoE) of Online Social Gaming," TU Delft, Delft University of Technology, 2014.
- [100] C. Zhang, "Quality of services and user experience in wireless vehicular internet access," The Hong Kong Polytechnic University, 2015.
- [101] T. A. Q. Pham, K. Piamrat, and C. Viho, "QoE-Aware Routing for Video Streaming over VANETs," in *Proceedings of the 2014 IEEE Vehicular Technology Conference (VTC-Fall)*, 2014, pp. 1–5.
- [102] M. Xing and L. Cai, "Adaptive video streaming with inter-vehicle relay for highway VANET scenario," in *Proceedings of the 2012 IEEE International Conference on Communications (ICC)*, 2012, pp. 5168–5172.
- [103] S. Jelassi, A. Bouzid, and H. Youssef, "QoE-Driven Video Streaming System over Cloud-Based VANET," in *Proceedings of the 2015 International Workshop on Communication Technologies for Vehicles: Nets4Cars/Nets4Trains/Nets4Aircraft (CTV)*, 2015, pp. 84–93.
- [104] C. Xu, F. Zhao, J. Guan, H. Zhang, and G. M. Muntean, "QoE-Driven User-Centric VoD Services in Urban Multihomed P2P-Based Vehicular Networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 5, pp. 2273–2289, 2013.
- [105] M. Fiedler, T. Hossfeld, and P. Tran-Gia, "A generic quantitative relationship between quality of experience and quality of service," *IEEE Network*, vol. 24, no. 2, pp. 36–41, 2010.
- [106] K. Brunnström, S. A. Beker, K. De Moor, A. Dooms, S. Egger, M.-N. Garcia, T. Hossfeld, S. Jumisko-Pyykkö, C. Keimel, M.-C. Larabi, B. Lawlor, P. Le Callet, S. Möller, F. Pereira, M. Pereira, A. Perkis, J. Pibernik, A. Pinheiro, A. Raake, P. Reichl, U. Reiter, R. Schatz, P. Schelkens, L. Skorin-Kapov, D. Strohmeier, C. Timmerer, M. Varela, I. Wechsung, J. You, and A. Zgank, "Qualinet White Paper on Definitions of Quality of Experience." Qualinet for publishing, 2013.
- [107] uber, "Uber publication on safety <<https://www.uber.com/safety/>>." uber, Aug-2016.
- [108] M. Aloqaily, B. Kantarci, and H. T. Mouftah, "On the impact of quality of experience (QoE) in a vehicular cloud with various providers," in *Proceedings of the 2014 High Capacity Optical Networks and Emerging/Enabling Technologies Photonics for Energy (HONET)*, 2014, pp. 94–98.
- [109] M. Aloqaily, B. Kantarci, and H. T. Mouftah, "A Generalized Framework for Quality of Experience (QoE)-based Provisioning in a Vehicular Cloud," in *Proceedings of the 2015 IEEE International Conference on Ubiquitous Wireless Broadband (ICUWB)*, 2015, pp. 1–5.
- [110] C. Bettstetter, G. Resta, and P. Santi, "The node distribution of the random waypoint mobility model for wireless ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 2, no. 3, pp. 257–269, 2003.
- [111] B. Kantarci, L. Foschini, A. Corradi, and H. T. Mouftah, "Design of energy-efficient cloud systems via network and resource virtualization," *International Journal of Network Management, Wiley Online Library*, vol. 25, no. 2, pp. 75–94, 2015.

- [112] H. Takabi, J. B. Joshi, and G.-J. Ahn, "Security and privacy challenges in cloud computing environments," *IEEE Security and Privacy*, no. 6, pp. 24–31, 2010.
- [113] B. Kantarci and H. T. Mouftah, "Economizing the operational costs of cloud services in an optical transport network," in *Proceedings of the 2013 International Conference on Transparent Optical Networks (ICTON)*, 2013, pp. 1–5.
- [114] B. Kantarci and H. T. Mouftah, "Minimizing the provisioning delay in the cloud network: Benefits, overheads and challenges," in *Proceedings of the 2012 IEEE Symposium on Computers and Communications (ISCC)*, 2012, pp. 806–811.
- [115] J. Suomalainen and J. Julku, "Enhancing Privacy of Information Brokering in Smart Districts by Adaptive Pseudonymization," *IEEE Access*, vol. 4, no. 1, pp. 914–927, 2016.
- [116] E. Fonseca, A. Festag, R. Baldessari, and R. L. Aguiar, "Support of Anonymity in VANETs - Putting Pseudonymity into Practice," in *Proceedings of the 2007 IEEE Wireless Communications and Networking Conference (WCNC)*, 2007, pp. 3400–3405.
- [117] D. Forster, F. Kargl, and H. Lohr, "PUCA: A pseudonym scheme with user-controlled anonymity for vehicular ad-hoc networks (VANET)," in *Proceedings of the 2014 IEEE Vehicular Networking Conference (VNC)*, 2014, pp. 25–32.
- [118] A. Lazarevic, L. Ertöz, V. Kumar, A. Ozgur, and J. Srivastava, "A Comparative Study of Anomaly Detection Schemes in Network Intrusion Detection," in *Proceedings of the 2003 SIAM International Conference on Data Mining (CDM)*, 2003, pp. 25–36.
- [119] G. Androulidakis, V. Chatzigiannakis, and S. Papavassiliou, "Network anomaly detection and classification via opportunistic sampling," *IEEE Network*, vol. 23, no. 1, pp. 6–12, 2009.
- [120] P. Xu, H. Jin, Q. Wu, and W. Wang, "Public-Key Encryption with Fuzzy Keyword Search: A Provably Secure Scheme under Keyword Guessing Attack," *IEEE Transactions on Computers*, vol. 62, no. 11, pp. 2266–2277, 2013.
- [121] R. Chen, Y. Mu, G. Yang, F. Guo, and X. Wang, "Dual-Server Public-Key Encryption With Keyword Search for Secure Cloud Storage," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 4, pp. 789–798, 2016.
- [122] Y. Kim, J. Kwak, and S. Chong, "Dual-side dynamic controls for cost minimization in mobile cloud computing systems," in *Proceedings of the 2015 International Symposium on Modeling and Optimization in Mobile Ad Hoc and Wireless Networks (WiOpt)*, 2015, pp. 443–450.
- [123] H. Yoo and D. Kim, "Repetition-based cooperative broadcasting for vehicular ad-hoc networks," *Journal of Computer Communications, Elsevier*, vol. 34, no. 15, pp. 1870–1882, 2011.
- [124] M. Torrent-Moreno, F. Schmidt-Eisenlohr, H. Fussler, and H. Hartenstein, "Effects of a realistic channel model on packet forwarding in vehicular ad hoc networks," in *Proceedings of the 2006 IEEE Wireless Communications and Networking Conference (WCNC)*, 2006, pp. 385–391.
- [125] ITU-T, "Definition of Quality of Experience (QoE)." ITU-T, May-2007.
- [126] M. Varela, L. Skorin-Kapov, and T. Ebrahimi, *Quality of Experience: Advanced Concepts, Applications and Methods*. Springer International Publishing, ISBN: 978-3-319-02681-7, 2014.

- [127] M. Aloqaily, B. Kantarci, and H. T. Mouftah, "An Auction-driven Multi-objective Provisioning Framework in a Vehicular Cloud," in *Proceedings of the 2015 IEEE GlobeCom Workshop on Cloud Computing Systems, Networks and Applications (CCSNA)*, 2015, pp. 1–6.
- [128] N. Yaagoubi and H. T. Mouftah, "User-Aware Game Theoretic Approach for Demand Management," *IEEE Transactions on Smart Grid*, vol. 6, no. 2, pp. 716–725, 2015.
- [129] W. Saad, Z. Han, A. Hjørungnes, D. Niyato, and E. Hossain, "Coalition Formation Games for Distributed Cooperation Among Roadside Units in Vehicular Networks," *IEEE Journal of Selected Topics in Communications*, vol. 29, no. 1, pp. 48–60, 2011.
- [130] Y. He, L. V. Sun, W. Yang, and H. Li, "A Game Theory-Based Analysis of Data Privacy in Vehicular Sensor Networks," *International Journal of Distributed Sensor Networks*, Hindawi Publishing Corporation, vol. 2014, no. 1, pp. 1–14, 2014.
- [131] W. Saad, Z. Han, H. V. Poor, and T. Basar, "Game-Theoretic Methods for the Smart Grid: An Overview of Microgrid Systems, Demand-Side Management, and Smart Grid Communications," *IEEE Signal Processing Magazine*, vol. 29, no. 5, pp. 86–105, 2012.
- [132] D. Ardagna, B. Panicucci, and M. Passacantando, "Generalized nash equilibria for the service provisioning problem in cloud systems," *IEEE Transactions on Services and Computing*, vol. 6, no. 4, pp. 429–442, 2013.
- [133] M. Aloqaily, B. Kantarci, and H. T. Mouftah, "Multiagent/Multiobjective Interaction Game System for Service Provisioning in Vehicular Cloud," *IEEE Access: Advances in Vehicular Clouds*, vol. 4, no. 1, pp. 3153–3168, 2016.
- [134] M. Pouryazdan, B. Kantarci, T. Soyata, and H. Song, "Anchor-Assisted and Vote-Based Trustworthiness Assurance in Smart City Crowdsensing," *IEEE Access: Smart Cities*, vol. 4, no. 1, pp. 529–541, 2016.
- [135] L. M. Vaquero, L. Roderó-Merino, J. Caceres, and M. Lindner, "A break in the clouds: towards a cloud definition," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 1, pp. 50–55, 2008.
- [136] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging {IT} platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Computer Systems*, Elsevier, vol. 25, no. 6, pp. 599–616, 2009.
- [137] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," *National Institute of Standards and Technology's (NIST), NIST*, pp. 1–7.
- [138] C. Gong, J. Liu, Q. Zhang, H. Chen, and Z. Gong, "The Characteristics of Cloud Computing," in *Proceedings of the 2010 International Conference on Parallel Processing Workshops (ICPPW)*, 2010, pp. 275–279.
- [139] D. Puthal, B. P. S. Sahoo, S. Mishra, and S. Swain, "Cloud Computing Features, Issues, and Challenges: A Big Picture," in *Proceedings of the 2015 International Conference on Computational Intelligence and Networks (CINE)*, 2015, pp. 116–123.

- [140] A. Botta, W. de Donato, V. Persico, and A. Pescapé, "Integration of Cloud computing and Internet of Things: A survey," *Future Generation Computer Systems*, Elsevier, vol. 56, pp. 684–700, Mar. 2016.
- [141] R. Woitsch and W. Utz, "Business Process as a Service: Model Based Business and IT Cloud Alignment as a Cloud Offering," in *Proceedings of the 2015 International Conference on Enterprise Systems (ES)*, 2015, pp. 121–130.
- [142] A. Giessmann and C. Legner, "Designing business models for cloud platforms," *Information Systems Journal: Digitization in Business Models and Entrepreneurship*, Wiley Online Library, vol. 26, no. 5, pp. 36–41, 2016.
- [143] L. Youseff, M. Butrico, and D. D. Silva, "Toward a Unified Ontology of Cloud Computing," in *Proceedings of the 2008 Grid Computing Environments Workshop (GCE)*, 2008, pp. 1–10.
- [144] B. Sotomayor, R. S. Montero, I. M. Llorente, and I. Foster, "Virtual Infrastructure Management in Private and Hybrid Clouds," *IEEE Internet Computing*, vol. 13, no. 5, pp. 14–22, 2009.
- [145] D. Nurmi, R. Wolski, C. Grzegorzczak, G. Obertelli, S. Soman, L. Youseff, and D. Zagorodnov, "The Eucalyptus Open-Source Cloud-Computing System," in *Proceedings of the 2009 IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID)*, 2009, pp. 124–131.
- [146] B. Wilder, *Cloud Architecture Patterns - Using Microsoft Azure*. O'Reilly Media, ISBN: 9781449319779, 2012.
- [147] W. Sun, X. Zhang, C. J. Guo, P. Sun, and H. Su, "Software as a Service: Configuration and Customization Perspectives," in *Proceedings of the 2008 IEEE Congress on Services Part II (SERVICES-2)*, 2008, pp. 18–25.
- [148] W. T. Tsai, X. Sun, and J. Balasooriya, "Service-Oriented Cloud Computing Architecture," in *Proceedings of the 2010 International Conference on Information Technology: New Generations (ITNG)*, 2010, pp. 684–689.
- [149] P. Sareen, "Cloud computing: types, architecture, applications, concerns, virtualization and role of it governance in cloud," *International Journal of Advanced Research in Computer Science and Software Engineering*, IJARCSSE online, vol. 3, no. 3, pp. 533–538, 2013.
- [150] N. Fernando, S. W. Loke, and W. Rahayu, "Mobile Cloud Computing: A Survey," *Future Generation Computer Systems*, Elsevier, vol. 29, no. 1, pp. 84–106, 2013.
- [151] F. Liu, P. Shu, H. Jin, L. Ding, J. Yu, D. Niu, and B. Li, "Gearing resource-poor mobile devices with powerful clouds: architectures, challenges, and applications," *IEEE Wireless Communications*, vol. 20, no. 3, pp. 14–22, 2013.
- [152] Mobile-Cloud-Computing-Organization-forum, "Discover the world of Mobile Cloud Computing," *MCC organization-forum London*, 2011.
- [153] L. Liu, R. Moulic, and D. Shea, "Cloud service portal for mobile device management," in *Proceedings of the 2010 IEEE International Conference on e-Business Engineering (ICEBE)*, 2010, pp. 474–478.
- [154] E. Cuervo, A. Balasubramanian, D. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, "MAUI: making smartphones last longer with code offload," in *Proceedings of the 2010 International conference on Mobile systems, applications, and services (MSAS)*, 2010, pp. 49–62.

- [155] U. Kremer, J. Hicks, and J. Rehg, "A compilation framework for power and energy management on mobile computers," in *Proceedings of the 2001 International Workshop on Languages and Compilers for Parallel Computing (LCPC)*, 2001, pp. 115–131.
- [156] K. Kumar and Y. Lu, "Cloud Computing For Mobile Users: Can Offloading Computation Save Energy?," in *Proceedings of the 2010 Purdue University, IEEE Computer Society*, 2010, pp. 51–56.
- [157] L. Guan, X. Ke, M. Song, and J. Song, "A Survey of Research on Mobile Cloud Computing," *Proceedings of the 2011 IEEE/ACIS International Conference on Computer and Information Science (ICIS)*, pp. 387–392, 2011.
- [158] S. S. Qureshi, T. Ahmad, K. Rafique, and S. ul-islam, "Mobile cloud computing as future for mobile applications-Implementation methods and challenging issues," in *Proceedings of the 2011 IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS)*, 2011, pp. 467–471.
- [159] D. Huang, "Mobile cloud computing," *IEEE COMSOC Multimedia Communications Technical Committee (MMTC) E-Letter*, vol. 6, no. 10, pp. 27–31, 2011.
- [160] U. Varshney, "Pervasive healthcare and wireless health monitoring," *Mobile Networks and Applications, Springer*, vol. 12, no. 2, pp. 113–127, 2007.
- [161] X. Yang, T. Pan, and J. Shen, "On 3G mobile e-commerce platform based on cloud computing," in *Proceedings of the 2010 IEEE International Conference on Ubi-media Computing (U-Media)*, 2010, pp. 198–201.
- [162] W. Zhao, Y. Sun, and L. Dai, "Improving computer basis teaching through mobile communication and cloud computing technology," in *Proceedings of the 2010 International Conference on Advanced Computer Theory and Engineering (ICACTE)*, 2010, pp. 452–454.

APPENDIX A CLOUD COMPUTING (CC) PARADIGM

The cloud computing paradigm is a development umbrella to define a category of sophisticated on-demand computing services offered by commercial providers such as Microsoft, Amazon and Google. The main objective of the cloud is to offer computing, service and storage to customers and corporations, and over the past few years several researchers have defined “cloud computing”. Vaquero et al. [135] described a cloud as a large pool of virtualized resources accessible via a pay-per-use scheme which can be reconfigured to ensure scalability. Buyya et al. [136] defined a cloud as a parallel distributed computing system based on service-level agreements (SLA); virtualized dynamic resource provisioning that can be presented as a unified computing resource between service providers and cloud users. More generally, Armbrust et al. [2] defined a cloud as a hardware and software data center that delivers on-demand services. And the National Institute of Standards and Technology (NIST) stated, “*Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models*” [137].

A.1 PROPERTIES OF THE CLOUD

The future of all communications systems will require cloud computing, and the abovementioned definitions have revealed common properties and novel characteristics [138][139][140], such as pay-per-use and no contract commitment. Since CC can provide many computational resources, as well as storage and services on-demand, cloud users will no longer need to plan ahead for infrastructure and resource provisioning. CC is also an adaptable, dynamic and scalable system that can be self-configured, and which provides unlimited capacity and an unlimited number of available resources. Such advancements, allow cloud users to rent the precise services they need.

Many enterprises benefit from and rely on this technology. For example, new start-ups can save considerably if they can choose to rent or borrow infrastructure and services based on the nature and period of their projects, rather than investing in an entire IT infrastructure in advance. Several studies [141][142] have proposed the idea of developing business processes as a cloud-based service, and in [141] the CloudSocket version of Business Processes as a Service (BPaaS) was presented. Similarly, a theory for platform business models was designed [142] to create a smart eco-system for cloud platform.

A.2 LAYERS OF CLOUD

As shown in Figure A.1, a cloud system offers three main levels of service: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) [1][10][25][143]. IaaS offers computation and communication resources, such as hardware, servers and storage [144], and the client is charged based only on their volume of usage, which can mean considerable savings. In addition, IaaS is very dynamic, and it can expand or decrease based on the number of users. Currently, the most popular examples of IaaS are Amazon's EC2 (Elastic Cloud Computing) and S3 (Simple Storage Service) [145].



Figure A.1: Layers of the cloud

PaaS is at a higher level of abstraction than IaaS, and it makes a cloud easy to program (as a plug-in) by offering advanced integrated environments for building, testing and deploying users' applications. Cloud users do not have to worry about the lower-level details of the environment, since the service providers handle all the storage, network and

deployment requirements. Google Apps Engine, Microsoft Azure and Amazon Maps are the best-known examples of PaaS [146].

SaaS is the highest level of abstraction in the cloud architecture. It allows users to share and access applications remotely via the Internet, by shifting the work of local computer programs to remote on-line software services with the same functionalities. Salesforce and Microsoft's Live Mesh are two examples of this service model [147]. The major advantage of SaaS is eliminating the need for compatibility between the software and the operating system, since users can access the required services via any available web browser.

Cloud users can use the available services in a flexible and efficient manner. In addition to the benefits of the abstraction levels, a cloud doesn't require building the three layers over each other directly [1]. For example, the SaaS level can be set up directly on top of IaaS, rather than PaaS. This advantage means extra flexibility for both users and developers.

In some special environments, a fourth level is added under the other cloud levels, namely *Data Centers (DC)*. Data centers are usually built in less populated areas to take advantage of cheaper energy rates and the reduced risk of natural disasters. This layer essentially provides the hardware facilities and infrastructure for the cloud [1][148].

A.3 CLOUD TYPES

There are three different types of clouds that can be deployed in different fields [1][2][10][149], namely *Public clouds*, *Private clouds* and *Hybrid clouds*. Public clouds offer services that are available for general use over the Internet, while private clouds are designated for a particular organization, and could include special services not offered by the public cloud. Hybrid clouds are combination of two or more private and public clouds, as shown in Figure A.2.

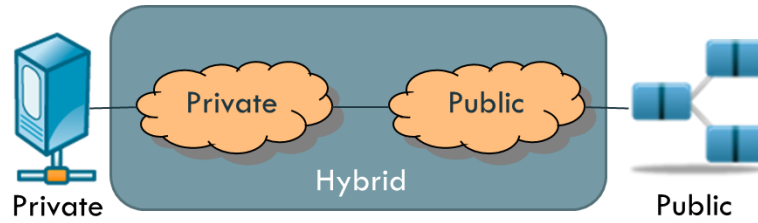


Figure A.2: Types of the cloud

A.4 MOBILE CLOUD COMPUTING (MCC)

Today, the new and expanding development of advanced mobile devices is an area that needs to be recognized and addressed. For example, smart phones are now fully equipped with a number of applications, including front and back cameras, smart sensors, high level processing cores, large screens, as well as far more memory and storage compared to previous models. These tools and advancements in mobile devices have enabled the explosion of the technology known as Mobile Cloud Computing (MCC). MCC consolidates cloud computing and mobile networking, and thus defines a new business model between mobile users, mobile network operators and cloud service providers. Mobile clouds offload the data storage and processing functions of a user's mobile device, a technique that saves considerable computing power and allows data storage that is not dependent on mobile devices, since it is resident in the cloud [1][150][151]. All user applications can then be easily accessed via a wireless connection and the web browser resident on the mobile devices.

The *Mobile Cloud Computing Forum* [152] is an organization that defines mobile cloud as an infrastructure in which both the data storage and processing functions are performed at the cloud, and not from the user side. This model of offloading the computing power and data storage requirements outside the mobile devices is a very promising development, as it will accommodate a far broader range of mobile subscribers.

MCC can be described as the offloading solution for mobile devices, in that it provides access to a substantial shared pool of resources that mobile users can exploit via a mobile backbone. The mobile cloud approach is currently the only tool that allows mobile users

to access applications and services on the Internet. The users' mobile devices do not need to have high capacity or powerful processing resources, since all the computing intensive services that require high capacity storage are handled at the cloud [153][154].

The general architecture of MCC can be seen in Figure A.3. The left side of the figure shows the front-end mobile devices that can access the Internet via a mobile backbone, and illustrates how service requests and other relevant information can be transmitted to the service provider via the backbone.

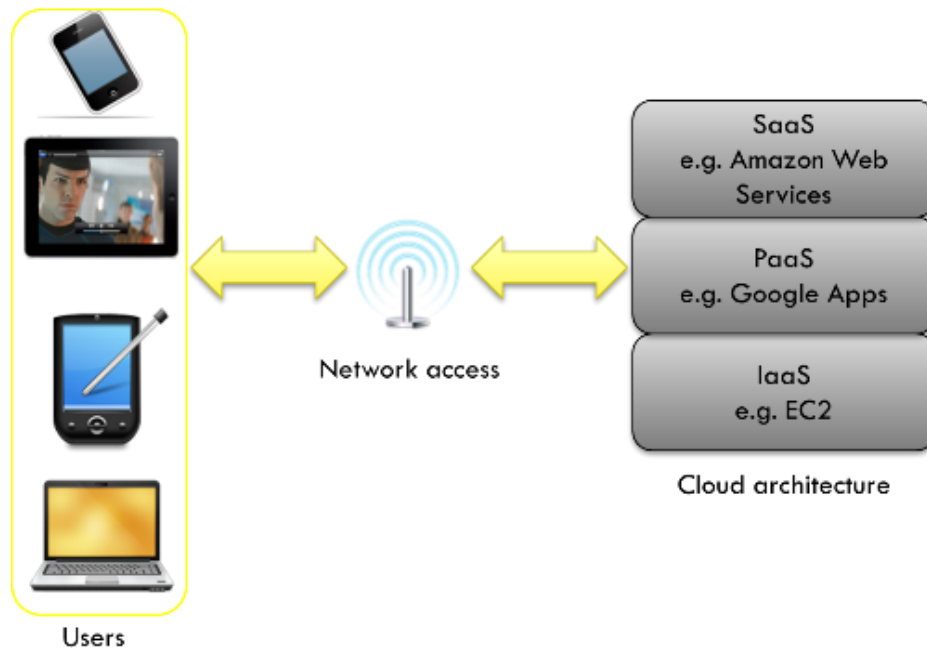


Figure A.3: Mobile cloud computing model

A.5 BENEFITS OF MOBILE CLOUD COMPUTING

As MCC becomes more attainable, market predictions and investments are increasingly moving toward this technology, since it has all the benefits of CC and is recognized as a promising solution for most mobile computing issues. This reasoning is due to the following distinct arguments.

(1) *Flexibility*: MCC is flexible and dynamic, and will allow users to run their mobile applications with no shortage of resources. These qualities also allow mobile users to run

their applications without the need for advance planning, and service providers can choose to arrange their applications using a minimum of resources, then scale up if required. Figure A.4 shows the different layers of interactions and access between cloud layers and users.

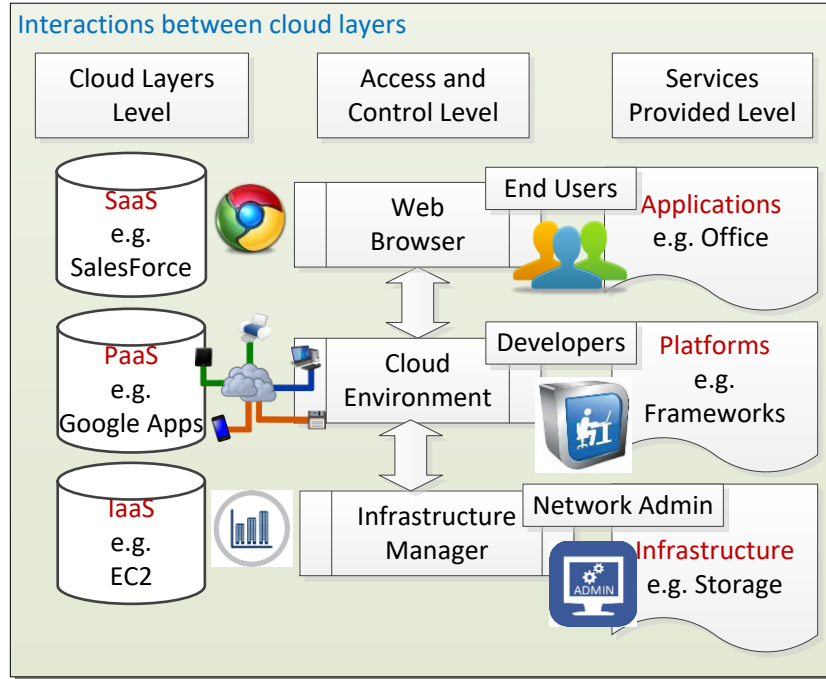


Figure A.4: Interactions and access of cloud layers

(2) *Battery lifetime:* The battery lifetime of mobile devices will be extended, since all computations happen at the cloud. Several traditional methods have been proposed to extend the battery lifetime of mobile devices, including enhancing mobile CPU performance, managing the screen brightness using sensors, and using dynamic smart cleaning for the internal disk. However, in the literature several studies [154][155][156][157] have evaluated the effectiveness of offloading computation power to a cloud, and the results indicate that using a cloud to improve the lifetime of users' mobile device batteries can significantly save energy.

(3) *Scalability:* User's demands will be more efficiently deployed, due to the flexible resource expansion of the cloud.

(4) *Integration*: Different service providers can combine several services, then provide the package to end mobile users.

(5) *Elimination of upfront commitment*: No advance planning for resource provisioning is required.

(6) *Ubiquity*: MCC services can be accessed from anywhere at any time.

(7) *Other benefits*: User friendly access, self-service, reliability and more.

Table A.1: Summary of CC, MCC and VCC main features

<i>Aspect</i>	<i>Description</i>	<i>CC</i>	<i>MCC</i>	<i>VCC</i>
<i>Lifetime Battery</i>	Limitation of the battery lifetime	No	Yes	No
<i>Computation</i>	Strength of computation	High	Low	High
<i>On-demand applications</i>	Applications are always available	Yes	On-demand	On-demand
<i>Pay as you go</i>	Cost per usage time	Yes	On-demand	On-demand
<i>NaaS,</i>	Support communications,	Yes	No	Yes
<i>SaaS,</i>	Support storage	Yes	No	Yes
<i>CaaS.</i>	Support information	On-demand	On-demand	Yes
<i>Traffic management</i>	Dynamic control of large traffic	No	No	Yes
<i>Disaster management</i>	Dynamic control of resources	On-demand	On-demand	Yes
<i>Mobility</i>	Availability of services while moving	No	Yes	Yes

A.6 APPLICATIONS OF MOBILE CLOUD COMPUTING

The increasing popularity of handheld mobile devices such as smartphones and tablets has introduced a new set of applications that can be deployed from a cloud. The smart devices have been used as data providers for different applications to save time and cost,

and many mobile system applications take advantage of MCC [1][158][159]. Some of these are as follows:

- 1) *Mobile Healthcare*: Mobile cloud-based healthcare will resolve many of the issues of legacy medical treatments. It will make several medical services (e.g. patients' medical records, patients' old treatments, etc.) available on-demand, within the same medical facility or around the globe. In addition, Mobile healthcare applications could provide hospitals, medical centers, medical laboratories and healthcare organizations a wide selection of on-demand services [160].
- 2) *Mobile Commerce*: Mobile commerce requires high mobility usage to perform on-demand tasks. Mobile commerce cloud computing will increase ongoing data processing and enhance security for mobile transactions, such as payments and online trading [161].
- 3) Many other mobile applications can take advantage of the benefits of the cloud, including *mobile e-learning* [162] and *mobile gaming*. For example, in [162] the authors present a holistic study of combining mobile learning systems and cloud computing to enhance the quality of communication between students and teachers.

APPENDIX B CONFIDENCE INTERVALS

The literature contains many quantitative methods to calculate result's accuracy. The most popular technique is Confidence Interval (CI). CI is a measure to quantify uncertainty over any collected sample of data. It is defined as the estimated range of values within which a generated data lies with a specific probability. For instance, a probability of 95% implies that a confidence of 95% that the collected data lies in a certain “confidence” interval. Simulated metrics such as end-to-end delays, availability and reliability are measured by calculating the mean of a successive of n runs, with different simulation seed to ensure that there no correlation in the presented results. All simulation runs have the same simulation environment although independent from each other.

As an example, the collected result for the Service Latency (SL) is considered, where the n independent results are represented by $SL_1, SL_2, \dots, SL_n$, where SL_i represents the Service Latency obtained from simulation run i . The mean of all the Service Latency simulation measurements can be expressed by:

$$\overline{SL} = \frac{1}{n} \sum_{i=1}^n SL_i \quad (\text{B.1})$$

However, the mean of the independent simulation runs SL provides a single numerical value for the estimate of the expected value of $E[SL] = \mu_s$. In order to evaluate the correctness of the estimate provided by SL for the simulation results, it is essential to compute the variance (σ_s^2). The variance can be calculated using the following equation:

$$\sigma_s^2 = \frac{1}{n-1} \sum_{i=1}^n (SL_i - \overline{SL})^2 \quad (\text{B.2})$$

Small (σ_s^2) indicates that the results are tightly clustered around $\overline{SL}$ and we can be confident that $\overline{SL}$ is close to the $E[SL]$. In case if (σ_s^2) is large, then, the results are widely dispersed around and we are less confident that $\overline{SL}$ is close to $E[SL]$. Instead of

seeking a single value to estimate the $E[SL]$, we can specify an interval of values that is highly likely to contain the true value of the parameter. Thus, a probability of $1 - \sigma_s$ is defined, an interval $[L(SL), U(SL)]$ is found such the probability is given by:

$$P[L(SL) \leq \mu_s \leq U(SL)] = 0.95 \quad (B.3)$$

This interval contains the true value of the parameter with probability of 0.95. Such an interval is a 95% CI. Using Standard deviation and t-distribution table (since the number of measurements used is less than 30), the lower and upper limits of the CI are calculated using the following two equations:

$$L(SL) = \overline{SL} - \frac{t_{[\frac{0.05}{2}, df]} * \sigma_s}{\sqrt{n}} \quad (B.4)$$

$$U(SL) = \overline{SL} + \frac{t_{[\frac{0.05}{2}, df]} * \sigma_s}{\sqrt{n}} \quad (B.5)$$

where n is the number of measurements, df is the degree of freedom and is equal to $n-1$,

and σ_s is the standard deviation of the measurements. In all of the simulations presented in this thesis the CI is calculated based on different runs. Through t-distribution value and Equations (B.4) and (B.5), it is found that the collected results were within the calculated value of CI (95%).