Design and Performance Evaluation of Service Discovery Protocols for Vehicular Networks

Kaouther Abrougui

Thesis submitted to the

Faculty of Graduate and Postdoctoral Studies

In partial fulfillment of the requirements

For the **PhD degree in Computer Science**

Ottawa-Carleton Institution for Computer Science

Faculty of Graduate and postdoctoral Studies

University of Ottawa

The following publications by the author are relevant to this thesis:

Journal

1. K. Abrougui, A. Boukerche, R. W. Pazzi: Location-aided Gateway Advertisement and Discovery Protocol for VANets. IEEE Transactions on Vehicular Technology, volume 59, number 8, pages 3843–3858, year 2010.

2. K. Abrougui, A. Boukerche, R. W. Pazzi: Design and Evaluation of Context-aware and Location-based Service Discovery Protocols for Vehicular Networks: IEEE Transactions on Intelligent Transportation Systems, volume: 12, issue: 3, pages: 717 – 735, year 2011.

3. K. Abrougui, A. Boukerche: Design and Performance Evaluation of an Efficient Fault Tolerant Location Based Service Discovery Protocol for Vehicular Networks. Accepted in the Journal of Network and Computer Applications 2011.

4. K. Abrougui, A. Boukerche, R. W. Pazzi: Scalable Bandwidth-Efficient Hybrid Adaptive Service Discovery Protocol for Vehicular Networks with Infrastructure Support. Submitted to IEEE Transactions on Mobile Computing.

5. K. Abrougui, A. Boukerche: Efficient Load Balancing and QoS based Location aware Service Discovery Protocol for VANets. Submitted to Eurasip Journal on Wireless Communication and Networking.

6. K. Abrougui, A. Boukerche, R. B. Araujo: Service Discovery Protocols: Challenges and Issues. To be submitted to IEEE Wireless Communication Magazine.

7. K. Abrougui, A. Boukerche: Service Discovery protocols: A Survey. To be submitted to Elsevier, Computer Networks.

Conference

1. K. Abrougui, R. W. Pazzi, A. Boukerche: Towards a Balanced and Reliable Localization of Services in Heterogeneous Vehicular Ad Hoc Networks. Accepted in the 7-th ACM Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet 2011).

2. K. Abrougui, A. Boukerche and R.W. Pazzi. Context-aware and location-based service discovery protocol for Vehicular Networks: Proof of correctness. Conference on Local Computer Networks (LCN), 2010 IEEE 35th pages 858–865.

3. K. Abrougui, A. Boukerche, R. W. Pazzi: An Efficient Fault Tolerant Location Based Service Discovery Protocol for Vehicular Networks. IEEE Global Telecommunications Conference: GLOBECOM 2010.

4. R. W. Pazzi, K. Abrougui, C. De Rezende, and A. Boukerche: Service Discovery Protocols for VANET based Emergency Preparedness Class of Applications: A Necessity Public Safety and Security, 4th International Conference on Information Systems Technology and Management (ICISTM 2010), March 11-13, 2010 Bangkok.

5. K. Abrougui, A. Boukerche, R. W. Pazzi: Performance Evaluation of Location-based Service Discovery Protocols for Vehicular Networks. The IEEE International Conference on Communications, 2010. ICC '10. ICC 2010.

6. A. Boukerche, K. Abrougui, R. W. Pazzi: Context-aware and Location-based Service Discovery Protocol for Vehicular Networks. International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems. Proceedings of the 6th ACM symposium on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks. PE-WASUN 09. Pages 93-100.

7. A. Boukerche, K. Abrougui, R. W. Pazzi: Location-aided Gateway Advertisement and Discovery Protocol for VANets: Proof of Correctness : IEEE 34th Conference on Local Computer Networks. LCN 2009. Page(s): 778 - 785.

8. A. Boukerche, K. Abrougui, R. W. Pazzi: An Efficient Hybrid Adaptive Location-Aided Gateway Advertisement and Discovery Protocol for Heterogeneous Wireless and Mobile Networks. . IEEE Global Telecommunications Conference, 2009. GLOBECOM 2009. Page(s): 1 - 6.

9. A. Boukerche, K. Abrougui: A Service Discovery protocol for vehicular ad hoc networks: A proof of correctness. IEEE International Symposium on Parallel and Distributed Processing, 2009. IPDPS 2009. Page(s): 1 - 8.

10. A. Boukerche, K. Abrougui: Performance evaluation of a hybrid adaptive service discovery protocol for next generation Heterogeneous Vehicular Networks (HVNs). International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems. Proceedings of the 3nd ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks. PM2HW2N 2008: 27-34.

Abstract

Intelligent Transportation Systems (ITS) are gaining momentum among researchers. ITS encompasses several technologies, including wireless communications, sensor networks, data and voice communication, real-time driving assistant systems, etc. These states of the art technologies are expected to pave the way for a plethora of vehicular network applications. In fact, recently we have witnessed a growing interest in Vehicular Networks from both the research community and industry. Several potential applications of Vehicular Networks are envisioned such as road safety and security, traffic monitoring and driving comfort, just to mention a few. It is critical that the existence of convenience or driving comfort services do not negatively affect the performance of safety services. In essence, the dissemination of safety services or the discovery of convenience applications requires the communication among service providers and service requesters through constrained bandwidth resources. Therefore, service discovery techniques for vehicular networks must efficiently use the available common resources.

In this thesis, we focus on the design of bandwidth-efficient and scalable service discovery protocols for Vehicular Networks. Three types of service discovery architectures are introduced: infrastructure-less, infrastructure-based, and hybrid architectures. Our proposed algorithms are network layer based where service discovery messages are integrated into the routing messages for a lightweight discovery. Moreover, our protocols use the channel diversity for efficient service discovery. We describe our algorithms and discuss their implementation. Finally, we present the main results of the extensive set of simulation experiments that have been used in order to evaluate their performance.

Acknowledgements

I would like to give special thanks to my supervisor Dr. Azzedine Boukerche for his financial support, his confidence in my abilities, his stimulating suggestions and encouragement that helped me to complete this work.

I would like to thank Dr. Richard Pazzi, from PARADISE Research Laboratory at the University of Ottawa and the NSERC DIVA Program Manager, for his continuous help during this work.

I would like to thank the NSERC organization and the NSERC DIVA Network for providing me their financial support and trusting my abilities.

I would like to thank all my friends in the PARADISE Research Laboratory for their support and companion during all these years.

I would also like to thank my parents Mohamed Najib Abrougui and Zohra Beji, my older sister Aida Abrougui and her husband Mourad Elhadef, and my younger sister Abir Abrougui for their continuous support and constant encouragement. I dedicate this thesis to my parents Najib and Zohra, to my daughter Sara, and to her father Racha Ben Ali.

I would like to thank all my friends and relatives for their encouragement and help.

Contents

List of Figures

List of Tables

Glossary of Terms

AODV Ad Hoc On-Demand Vector Routing

CCA Cooperative Collision Avoidance

CLA-S Connection Less Approach for Streets

DSDV Destination Sequenced Distance Vector

DHT Distributed Hash Table

DREAM Distance Routing Effect Algorithm for Mobility

DSR Dynamic Source Routing

EB-LocVSDP Election based LocVSDP

FTLocVSDP Fault Tolerant LocVSDP

GPD Gateway Proactive Discovery

GRD Gateway Reactive Discovery

GWSOL Gateway Solicitation

GHT Geographic Hash Table

GPS Global Positioning System

IGS Information Guided Search

I-BIA Intelligent Broadcast with Implicit Acknowledgement

LAGAD Location Aided Advertisement and Discovery

LAR Location-Aided Routing (LAR)

L-VPD Location-based Vehicular Proactive Discovery

L-VRD Location-based Vehicular Reactive Discovery

LocVSDP Location-based Vehicular Service Discovery Protocol

MAC Medium Access Control

MANet Mobile Ad hoc Network

NB Naive Broadcast

NDP Neighbor Discovery Protocol

NBID Neighbor Identifier

NS2 Network Simulator 2

PMNET Parking Meter Network

PPB Peer-to-Peer Broadcast

PAZ Proactive Advertisement Zone

QoS Quality of Service

QoSLocVSDP Load-Balancing and QoS-Aware LocVSDP

RI The Region of Interest

SLP Service Location Protocol

SOTIS Self-Organizing Traffic Information System

SHARP Hybrid Adaptive Routing Protocol for Mobile Ad hoc Networks

SPR SHARP Proactive Routing

TORA Temporally Ordered Routing Algorithm

TDMA Time Division Multiple Access

TTL Time to Live

UDP User Datagram Protocol

VAHS Virtual Ad-Hoc Server

VANets Vehicular Ad hoc Networks

VCWC Vehicular Collision Warning Communication

VITP Vehicular Information Transfer Protocol

VPD Vehicular Proactive Discovery

VRD Vehicular Reactive Discovery

VSDP Vehicular Service Discovery Protocol

VAHS Virtual Ad-Hoc Server

VMN Virtual Mobile Node

Chapter 1

Introduction

1.1 Background

Recently, Vehicular Ad hoc Networks (VANets) [15] have been gaining great interest from the research community. They are composed of a large number of vehicular components to permit the communication and the dissemination of information between vehicles. This type of network has become popular due to its applicability to several areas such as safety, environmental, meteorologic, informative, and entertainment. Despite the fact that the main goal of a vehicular network is to guarantee security on roads, several secondary objectives, or prerequisites, have to be achieved in order to reach the main objective. In fact, applications in vehicular networks heavily depend on service discovery for various reasons such as determining the locations of accidents and disasters and user guidance to alternatives roads; automotive diagnosis and user warning; determining location and proximity of restaurants, gas stations, service stations, and free parking spots; sharing music and files.

That is why, the definition of a service discovery system [25] among the vehicular nodes is one of the prerequisites needed in order to make viable many of the vehicular networks applications. The service discovery problem consists of finding services that satisfy a driver's or passenger's request. Such a problem is very ample and extensive and it has been investigated in many areas in wired networks, ad hoc networks, mesh networks. However, very little work has been done in VANets. At first glance, it is easy to state that service discovery in ad hoc networks [14, 37, 28, 49, 24, 76, 51, 85, 47, 39, 23, 73, 61, 66, 22] can be applied directly to vehicular networks. However, although vehicular networks have many similarities with mobile ad hoc networks such as their short radio transmission range, their low bandwidth and their omni-directional broadcast in most cases, they differentiate from ad hoc networks in many aspects. In fact, VANets are characterized by (i) their quick changes in link topology due to the high mobility speed of vehicles, (ii) frequent disconnections between vehicles mainly due to the low density of vehicles, (iii) limited bandwidth of wireless medium. Thus, data compression and aggregation is required. Moreover, in VANets it is possible to predict vehicle movements due to static roads and knowledge about the position and the speed of vehicles, vehicles are not energy-constraint as opposed to wireless nodes in ad hoc networks, services in vehicular networks are very important to drivers mainly in emergency situations, etc.

Due to the unique characteristics of vehicular networks, existing service discovery protocols in ad hoc networks cannot be applied directly to vehicular networks. In this thesis, the service discovery problem will be addressed under the viewpoint of the vehicular networks. In fact, service discovery systems have been identified as a key technology to the development and operation of the vehicular network [25]. Although their use is not exclusive to these networks, the service discovery systems differentiate from the other areas because their applicability have been greatly extended with the advent of the vehicular networks.

1.2 Motivation and Research Challenges

A mechanism is needed to allow drivers and passengers requesting services, and/or road components providing services, discover each other within the vehicular network. A driver or passenger may need a particular service that may be provided by one or more vehicles or roadside components. The driver needs a mechanism to find out which vehicular component provides the service desired by the vehicular requester. For the sake of simplification, we will refer to the driver requesting a service as a vehicular service requester. We will refer to the component (vehicle or roadside component) that provides the service as the vehicular service provider. The discovery mechanism needs to work without any prior configuration. For example, vehicles circulating on the roads and forming a vehicular network, should not be required to configure the discovery mechanism. They should be able to discover services on the vehicular network like free parking spots, localize restaurants or gas stations and get safety related information or guidance, etc.

The vehicular network imposes a number of prerequisites that must be taken into consideration by a service discovery mechanism, due to the particular mobility model and the importance of applications in vehicular networks. In the following, we cite the most important ones:

- *Auto-configuration*, which permits vehicles to form a vehicular network which will be able to use the discovery mechanism without any prior configuration.

- *Scalability*, where the discovery mechanism can be applied to large-scale and/or dense vehicular networks.

- *Efficiency*, while using the network resources because the discovery mechanism coexists with other communication protocols and needs to share the network resources.

- *Robustness*, which consists of tolerance to communication problems and capacity of continuing the provision of services to requesters in case of initial service providers' failures or network congestion.

- *Load balancing*, which assigns under utilized resources in loose regions to congested regions and thus, prevents the drawbacks associated with communications in highly congested regions.

- *Security*, by introducing a minimum level of security in order to prevent from erroneous service information dissemination in the vehicular network.

1.3 Objective and Contributions of Thesis

This thesis contributes with a suite of scalable service discovery protocols for VANets. Our protocols allow the drivers and passengers to discover a large set of services and allow service providers to be reached by a large set of drivers. Using these protocols, we shall propose various mechanisms to cover aspects related to how service requesters can reach service providers in large scale and highly mobile VANets. We shall also answer several questions related to how the success rate of service queries could be increased while the network overhead is decreased. The main contributions of this thesis are the design and development of:

1. A scalable location-aided gateway advertisement and discovery (LAGAD) protocol for VANets that does not rely on any infrastructure. LAGAD protocol permits gateway clients to continue being updated with the route toward the gateway. In other words, given a set of gateways and assuming that each road component is aware of its position, the goal of this protocol is to let each gateway requester car discover nearby gateways and gain sufficient information to route packets towards the closest gateway while guaranteeing the network scalability.

2. A scalable service discovery protocol for vehicular networks (VSDP) with infrastructure support which can work under high mobility models. VSDP is a novel service discovery technique dedicated for large scale vehicular networks. We also propose an efficient service providers' advertisement zone size adaptation scheme. This latter permits the adjustment and the prediction of the size of the advertisement zone of service providers through our proposed adjustment and prediction mechanisms, respectively. In VSDP, we combine proactive and reactive service discovery strategies to come up with an efficient and scalable hybrid adaptive service discovery protocol for vehicular networks.

3. A scalable location-aware service discovery protocol (LocVSDP) for vehicular networks that relies on a cluster-based infrastructure. LocVSDP finds services located in the region of interest specified in the driver's or passenger's request using an efficient

location-based propagation of the request and an efficient computation of the reply.

4. Fault tolerance mechanisms in order to enhance the performance of LocVSDP (FTLocVSDP). FTLocVSDP works well under (i) service providers' failures, (ii) Roadside Routers' failures, and (iii) communication links' failures.

5. Load balancing and QoS techniques to enhance further the performance of LocVSDP (QoSLocVSDP). The QoSLocVSDP balances the traffic load between road components and guarantees the satisfaction of drivers' QoS requirements.

Our proposed protocols are network layer based, where service discovery messages are integrated into the routing messages for a lightweight discovery. Moreover, they are multi-interfaces and multi-channels, which permit (i) an efficient usage of the radio spectrum, (ii) an increase of the wireless network capacity, and (iii) a delay reduction in service queries. For each proposed protocol in this thesis, we first present its formal description. Second, we discuss its proof of correctness. Third, we compute its complexities and cost functions. Finally, we present its performance evaluation after we conduct an extensive set of simulation experiments with the NS2 [4] and we compare it to related work.

1.4 Thesis Outline

This thesis comprises six chapters. They are organized as follows: Chapter 2 reviews existing service discovery protocols in VANets. Chapter 3 introduces the location aided gateway advertisement and discovery protocol for VANets. Chapter 4 presents the infrastructure based service discovery protocol for VANets. Chapter 5 presents the location aware service discovery protocol for VANets. We also describe and discuss the fault tolerance, load balancing and QoS features to enhance the performance of the discovery protocol. Chapter 6 concludes this thesis, and presents future works.

Chapter 2

Literature Review

2.1 Introduction

Through this chapter we will review the key issues related to this thesis. As the main focus of this thesis is to propose scalable service discovery protocols for vehicular networks. Two different aspects will be covered: vehicular networks, and service discovery protocols for vehicular networks. In the vehicular network part in Section 2.2, we will highlight the key characteristics and the main applications of vehicular networks. Since the service discovery problem is very important in our research, we will review the main approaches in the literature that have been proposed to solve the problem of service discovery over vehicular networks in the service discovery protocols'part in Section 2.3. In Section 2.4, we provide a classification of existing service discovery protocols in VANets according to their discovery mode. Finally, in Section 2.5, we summarize the chapter.

2.2 Vehicular Networks

The main purposes of vehicular networks are to provide safety along the road and to ensure the comfort of passengers by providing them the information and entertainment they require. Thus, two major types of vehicular networks applications are commonly known: safety applications and convenience applications. The former applications deal with the enhancement of vehicle safety on roads, while the latter help provide services to passengers in their vehicles.

2.2.1 Discovery of Safety Services

Safety on the roads on vehicular networks heavily depends on service discovery and advertisement for various reasons such as discovery of collisions and obstacles on the road and avoiding eventual accident [13]. Car crashes are one of the most common traffic accidents. According to the National Highway Traffic Safety Administration there are about 43,000 people killed in fatal car accidents each year in the United States [1]. Different factors can cause car crashes, such as the mechanical malfunctioning of the vehicle. It has been noticed on the statistics that the most important cause of car crashes is related to the driver behavior. In fact, if a driver cannot react on time to an emergency situation, a potential collision chain can be created on the road involving vehicles following the initial vehicle. In order to avoid collisions, a driver relies on the break lights of the vehicle immediately ahead. However, if vehicles are following each other very closely, drivers may not have enough time to react and to avoid collisions. A better way to perform collision avoidance would be to have an efficient *Collision warning services* (as in Figure 2.1A) that would provide emergency discovery and advertisement protocols to reduce the delay of propagation of collision warnings compared to the traditional reaction to the break lights of the leading vehicle.

Another reason for car crashes is related to the bad weather. In fact, a nasty weather that hammers the streets with snow, freezing rain, rain, fog, etc, could leave the roads in a messy and dangerous situation: Roads are slippery, the visibility is reduced, the road lines and even the road signs are hidden mainly during the night. In these severe conditions, vehicle crashes and slippage become high if the drivers are not informed accurately about the quality of the road and the visibility in the coming area. Drivers can slow down and be more attentive or even change their way if the condition on the front road is very dangerous or there are vehicles crashes blocking the road. A better way to help drivers overcome these situations and to avoid collisions and accident would be to have *(i) Environmental monitoring services* (as in Figure 2.1B) with installed sensor on vehicles to monitor environmental conditions, such as temperature, humidity, voltage and other factors; and *(ii) Civil infrastructure monitoring services* (as in Figure 2.1C) by installing sensors on vehicles to monitor the state of roads and detect slippery roads, etc. Drivers can discover these services either passively or reactively and can take the appropriate decision or be more alert and preventive before it is too late.

Safety on the road is also improved by discovering the *Automotive diagnostics services* (as in Figure 2.1D). In this kind of service, information is obtained from the vehicle on-board sensors that can detect and monitor bad driving tendencies. These latter are caused either by a mechanical malfunctioning on the vehicle or by a misbehavior of the driver because he is drunk or he falls asleep. If such situation is discovered with this kind of service, an alert is sent to the driver and to the surrounding drivers in order to be more careful and to avoid potential accidents.

Safety on the roads is also improved with the existence of *(i) Disaster warning services* (as in Figure 2.1E) and *(ii) User guidance services in case of disasters* (as in Figure 2.1F). Disaster warning services can help to detect disasters like bridge breakage, volcano eruption, flooding, avalanche, hurricane, etc. With the help of *Environmental monitoring services*, *Civil infrastructure monitoring services*, and *Disaster warning services*, warnings can be sent to drivers in the affected region so that, drivers can take the appropriate action before they become close to the affected region. On the other hand, drivers that are already very close to the disaster can use the *User guidance services in case of disasters* in order to be rescued and evacuated from the affected region. Road monitoring (in Figure 2.1G) helps drivers in critical situations to be discovered by the closest service station and get the required help, mainly if the driver is in an isolated region (for example, a driver who is out of fuel in the middle of an isolated road). From these examples, we conclude that it is clear that service discovery and advertisement is very important in order to improve the safety on the roads and prevent eventual accidents and deaths on the roads.

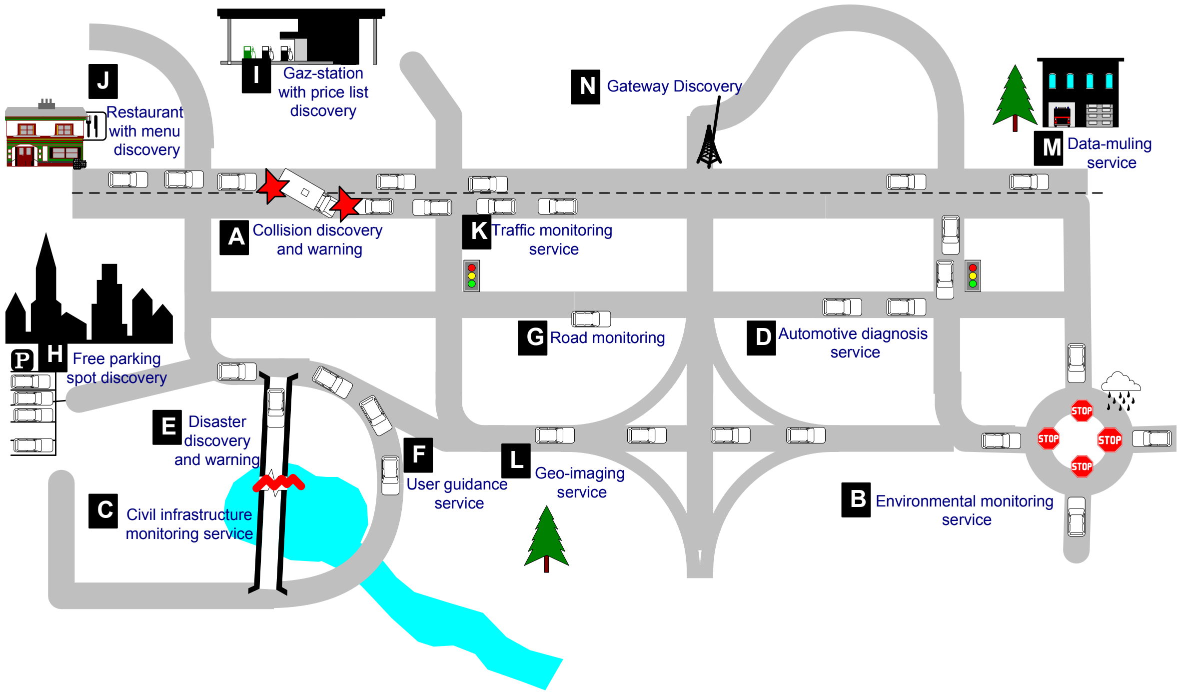


Figure 2.1: Service discovery in VANet

2.2.2 Discovery of Convenience Services

Convenience services could be entertainment services or informative services. Even if these types of services are not related to safety on roads, they are still important in vehicular networks. Let us consider the *Free parking spots service discovery* [18] (illustrated in Figure 2.1H). This type of service is considered as a convenience service that helps the driver to find a free parking spot in a congested city. Some studies have been performed in a congested city that highlight the annual damage caused by the parking space traffic search. This problem causes economical damages, gasoline and diesel wasted in search of free parking spots, wasted time for drivers, in addition to encumbering the entire traffic on the city [18]. Thus, an efficient *Free parking spots service discovery* protocol would overcome all of the problems mentioned earlier.

Another type of convenience service discovery consists on *Location discovery* where a driver would like to discover the gas-stations (illustrated in Figure 2.1I) on the road with their price list, in order to be able to choose the gas-station with the lowest price and take the appropriate exit. Another scenario could be the discovery a the nearby restaurants with their menu (illustrated in Figure 2.1J). The driver then can choose the preferred menu and takes the appropriate exit to the chosen restaurant.

The *Traffic monitoring service* (illustrated in Figure 2.1K) could be also discovered by drivers in order to help them to choose the less congested roads to their respective destinations.

Geo-imaging service discovery (illustrated in Figure 2.1L) is another type of convenience service that consists on attaching cameras on cars that capture images embedding data GPS coordinates automatically. These images are then transferred to the appropriate applications like the landmark-based routing and driving direction applications that could be installed on other cars or centralized for a general use.

Service discovery can also be used by data muling applications. In data muling, mobile agents interact with static agents in order to upload, download and transfer data to other physical locations. Thus, cars can be used to send data from remote sensornets to Internet servers. *Service discovery of data muling agents* (illustrated in Figure 2.1M) could help the communication and the interaction between the different agents. The *Gateway Discovery Service* (illustrated in Figure 2.1N) helps drivers when connecting to the Internet and retrieving information while they are in their cars.

2.3 Service Discovery in Vehicular Networks

In this section, we review existing service discovery protocols that have been investigated in the literature. Then we provide our classification of service discovery protocols depending on their mode. The following are the main service discovery and advertisement protocols in VANets:

1. ***CarTel System:*** The Cartel system [41] is a mobile sensor computing system comprised of CarTel nodes; in an automotive context, nodes are automobiles. Each node

comprises a computer coupled to a set of sensors. CarTel nodes permit to collect data from on-board and external sensors, process the gathered data and deliver it to a central portal for future use by the other CarTel nodes. The purpose of CarTel is to provide a simple programming interface that simplifies the development of mobile sensor network applications. The integration of new kinds of sensors, new mobile nodes and new data types is handled in CarTel. Moreover, CarTel handles intermittent connectivity. The CarTel system is composed of three main components: (i) the *portal*, which is considered as the central location to host applications related to CarTel and is responsible for controlling and configuring the distributed system; (ii) The *ICEDB*, which is an intermittently connected database, its main function is query processing and it tolerates delays; (iii) and *CafNet* which is related to carry and forward network and is a delay-tolerant network stack. If applications need data, they specify the properties of the data they want through continuous queries. These queries are executed on remote nodes using the delay-tolerant query processor ICEDB. The network stack CafNet permits to deliver data between the portal and the remote nodes even if the connectivity is intermittent.

2. ***MetroSense System:*** Campbell *et al.* proposed the MetroSense system [21]. MetroSense has a three-tier architecture: *(i)* the *Server Tier* comprising Ethernet-connected servers, *(ii)* the *Sensor Access Point Tier* permits secure gateway access to MetroSense services, and *(iii)* the *Sensor Tier* which comprises agents performing applications functions. MetroSense proposes a scalable architecture to support concurrent people-centric applications. In MetroSense project, there are investigation about virtual services in order to enhance the sensing and collection capabilities of sensor access points and mobile sensors. Mainly it consists on extending the physical sensing range of a device to cover an arbitrary area and is called virtual sensing. In this latter, *callers* or fixed nodes have the task to provide information about its surrounding environment. They announce their presence by sending beacons periodically. They run a selection algorithm to select the

best set of mobile nodes or *callees* to sense beyond their surrounding area. The selection of mobile nodes or callees in based on two factors: (i) the quality of links of mobile nodes; and (ii) the reading from a compass deployed in each node to know to which direction the mobile node is heading. The work on MetroSense project is focusing on people-centric sensing, in order to provide a new wireless sensor edge for the Internet.

3. ***UrbanSensing System:*** The UrbanSensing project [5] seeks to settle technological approaches for using embedded and mobile sensing for the benefit of the community. Many elements compose the UrbanSensing landscape such as: building, people, vehicles, etc. The UrbanSensing architecture is designed in such a way that permits to satisfy the low cost and scalability requirements. The system architecture of UrbanSensing comprise: *(i) Sensors* which are the sources of data at the edges of the network, they can be used also for configuring sensors and providing contextual information; *(ii) Subscribers* that can be either individual users or network applications; *(iii) Registries* that permit the discovery and the binding between sensors and subscribers, sensors register information about the data they publish with the registry, the subscribers search for their requested data using the registry; and *(iv) Mediators* which are network nodes that permit the selection of the requested data depending on a context function or any other function under the application control in order to enhance data streams.

4. ***The Decentralized Discovery of Free Parking Places:*** Caliskan *et al.* proposed a Decentralized Discovery of Free Parking Places [18]. The proposed protocol is not limited to free parking discovery, but can be applied to disseminate information with a spatio-temporal nature. Their protocol intends to inform drivers about free parking places in urban traffic conditions. Information about free parking places are disseminated to drivers in a proactive way. The information is broadcasted every predefined periodic interval by considering the spatio-temporal characteristics of the places. Caliskan *et al.* used in their model a central element, the *"automat"*, which is a term used in Germany to designate the terminal used for paying parking fees. In the protocol, the parking avail-

ability information could be either *(i)* atomic information or *(ii)* aggregate information. Atomic information provides parking availability information from one parking automat perspective, and they are distributed in a local proximity. Aggregate information provides parking availability information in an area covered by many parking automats, and they are distributed in wider areas. This can provide drivers an initial orientation about free parking places from a macro-perspective. Regions providing aggregate information are disjunct and organized hierarchically. Caliskan *et al.* used an overlay quad-tree structure to perform their aggregates. Aggregate information accuracy decreases with the increase of the distance separating a vehicle from the parking area. Caliskan *et al.* proposed a selection strategy for atomic and aggregate information in order to decrease the loss of accuracy. Their strategy is based on the relevance of atomic and aggregate information. Caliskan *et al.* proved that the level of redundancy in their protocol is very high. However, according to them, redundancy is desirable because it should help the distribution of information due to the highly expected partitioning of the vehicular ad hoc network. In their simulation, Caliskan *et al.* proved that the spatial distribution of atomic and aggregated information could be controlled through their selection strategy. This could guarantee efficient bandwidth utilization.

5. ***Warning Delivery Service for Inter-Vehicular Networks:*** Fracchia *et al.* proposed a Warning Delivery Service for inter-vehicular networks [29]. In their protocol, when a danger is detected, warning messages are propagated using multihop ad hoc communications. The purpose of their protocol is to guarantee that vehicles located in a predetermined area close to the danger, called *safety zone*, are informed. They used a highway traffic as a reference scenario where the high speed of vehicles has an impact on the timely delivery of the warning message. A vehicle in the safety zone that receives the warning message decides with a probability α to forward the message. However, a vehicle that is not located in the safety zone does not forward the message. A warning message contains the following information: *(i)* a warning identifier determined by the

original vehicle, *(ii)* the position of the danger and *(iii)* the time when the original vehicle starts the broadcast cycle. Vehicles store previously received messages in a list in order to be able to check the redundancy of received warning messages and avoid rebroadcasting the same ones. Fracchia *et al.* assume that the propagation of the warning messages is performed in a single traffic direction. The original vehicle starts new broadcast cycles every δt period along a period T in order to enhance the service reliability and robustness. Fracchia *et al.* implemented two different schemes for neighboring vehicles that receive a copy of the same warning message. In the first one, the receiving vehicle does not retransmit the copy of the same message. While, in the second scheme, the vehicle retransmits the copy only if a period h has elapsed from the previous reception of the copy. Fracchia *et al.* discussed the impact of h on the effectiveness of their warning delivery protocol. They adopted the carrier sense multiple access (CSMA) mechanism for the MAC protocol. They assumed a constant vehicle density along the highway.

6. ***TrafficView:*** Nadeem *et al.* proposed the TrafficView protocol [58]. TrafficView is a part of the e-Road project, which aims to provide a scalable infrastructure for inter-vehicle communication. TrafficView permits to vehicles to gather and disseminate traffic information. The main purpose of their protocol is to permit the dissemination of location and speed information between vehicles in order to permit to vehicles to assess traffic condition. The information is broadcasted periodically in TrafficView. TrafficView uses the *diffusion mechanism*, where every vehicle broadcasts information about itself and its neighboring vehicles for a purpose of scalability. Records in TrafficView are characterized by their small size, thus the protocol does not suffer from memory limitation. It is mainly designed for highway traffic scenario and based on road segment structure. Nadeem *et al.* in [58] implemented a prototype of TrafficView and the main components of their system are: *(i) the GPS/OBD module* that updates periodically the vehicle's own records (position and speed) in a validated dataset, *(ii) the receiver module* that collects records broadcasted from neighboring vehicles and stores them in the non-

validated dataset, *(iii) the validation module* that validates the non-validated dataset by applying some validation mechanism and merge them with the validated dataset, *(iv) the aggregation module* that permits to aggregate records in the validated dataset through an aggregation mechanism, *(v) the send module* that construct the broadcast message from the validated dataset and broadcasts it, and *(vi) the display/UI module* that displays the validated records on the display and permits the interaction with the driver.

7. ***Self-Organizing Traffic Information System:*** Wischhof *et al.* proposed SOTIS [79], a Self-Organizing Traffic Information System[1]. Their proposed system permits to vehicles to monitor traffic situations and distribute traffic messages. SOTIS does not rely on any infrastructure. Each SOTIS vehicle collects traffic information and processes it individually. The collected information is stored in a *Knowledge Base*. SOTIS is designed for highways and based on a road segment structure. Packets exchanged between SOTIS vehicles contain traffic information based on each road segment. The *Knowledge Base* contains traffic information of segments inside a specific area only. TDMA is used as the channel access technique in SOTIS. The traffic information outside the specific area are disregarded. SOTIS vehicles generate two types of reports: *(i) Periodic reports*, and *(ii) Emergency reports*. Periodic reports contain the individual knowledge of the SOTIS vehicle. The individual knowledge of a vehicle consists of the information of the current vehicle (such as its position, speed, etc.), and a short traffic situation analysis computed by the current vehicle based on its *Knowledge Base*. Periodic reports are broadcasted periodically by every SOTIS vehicle. Emergency reports are sent in the case of emergency. As opposed to periodic reports, emergency reports do not contain any traffic situation analysis, but only information about the type and the location of the emergency. Wischhof *et al.* used a reservation mechanism for the immediate access to the wireless channel in case of emergency. The *Knowledge Base* is updated upon reception of any type of the reports. In their simulation, Wischhof *et al.* used a *cellular*

[1] also part of the FleeNet [2] project.

automaton approach [59] allowing passing [60] for their traffic model. In their simulation, Wischhof *et al.* proved that traffic information can be provided to other vehicles with reasonable delay. For surrounding vehicles, traffic information are characterized by their high level of detail and low delay.

8. ***Reducing resource discovery time by spatio-temporal information in vehicular ad-hoc networks:*** Wolfson *et al.* proposed a competitive physical resource discovery protocol [80]. They introduced the *Peer-to-Peer Broadcast (PPB)* paradigm where every vehicle broadcasts its M most relevant available resources to its neighboring vehicles. The vehicle looking for a free parking place, keeps moving around the region of interest until it finds out about a free space through the *Information Guided Search (IGS)* mechanism proposed in the chapter. For their simulation, Wolfson *et al.* generated their own traffic model in a grid network. They proved through their experiments that by broadcasting the most relevant resource information, the bandwidth efficiency increases and vehicles are informed about available resources. However, the latency is high because vehicles have to move around the region of interest in order to find a free parking place.

9. ***Vehicle-to-Vehicle Wireless Communication Protocols for Enhancing Highway Traffic Safety:*** Biswas *et al.* proposed "Vehicle-to-Vehicle Wireless Communication Protocols for Enhancing Highway Traffic Safety" [13]. Biswas *et al.* proposed some mechanisms for *Cooperative Collision Avoidance (CCA)* in a highway through a class of context-aware forwarding protocols. In their protocol, the vehicle that detects the collision starts broadcasting a collision warning message. Biswas *et al.* proposed two direction-aware broadcasting mechanisms: *(i) Naive Broadcast (NB)* and *(ii) Intelligent Broadcast with Implicit Acknowledgement (I-BIA)*. In the first mechanism, the vehicle that detects the emergency starts broadcasting warning messages periodically. A vehicle in the same direction of the emergency that receives the warning message from behind, ignores the message. However, a vehicle in the same direction of the emergency that receives the warning message from the front, decelerates and broadcasts the warning

message. In this mechanism, Biswas *et al.* did not specify when the propagation of the warning message stops. The excessive forwarding of warning messages increases message collision in 802.11, thus delivery rate is decreased and delivery latency is increased. The second mechanism proposed by Biswas *et al.* in [13] overcomes the limitation of the first mechanism. In fact, in *I-BIA*, a vehicle that receives the warning message from behind stops the forwarding process, because it infers that at least one vehicle has received the warning message and will be responsible for forwarding it. Performance evaluation of their protocol showed that their cooperative collision avoidance system succeeded in reducing the number of collision in a platoon using the *NB* mechanism. The number of collisions is reduced more using the *I-BIA* mechanism. Biswas *et al.* in [13] studied the performance of the cooperative collision avoidance system when warning messages are given higher priority than background traffic. They proved that with data prioritization, more vehicles are prevented from collisions.

10. ***Vehicular Collision Warning Communication protocol:*** Yang *et al.* proposed a Vehicular Collision Warning Communication (VCWC) protocol [83]. The main purpose of VCWC is to provide warning messages to endangered vehicles as soon as possible such that eventual accidents could be avoided. The main design components of VCWC are as follow: *(i)* a message differentiation mechanism that enables safety and non-safety applications to share a common channel, *(ii)* congestion control policies that consists on a transmission rate decreasing algorithm and a state transition mechanism to eliminate warning messages redundancy while covering all the endangered region with a reasonable delivery delay, and *(iii)* warning dissemination methods to deliver warning messages in an efficient range. Their protocol achieves low latency and efficient bandwidth usage.

11. ***Gateway Discovery:*** Jelger *et al.* proposed a proactive gateway discovery approach [42]. Their protocol uses a prefix continuity property that ensures restricted flooding. In their approach, gateways send GW_INFO advertisements periodically. However, every ad

hoc node forwards only the messages needed to configure its own IP address, such that it has the same prefix as its next hop to the gateway. This approach helps to divide the network into many subnets, where the number of subnets is equal to the number of gateways in the MANet. The node that sends the GW_INFO to create or update the route to the gateway is referred to as the upstream neighbor. If this gateway discovery approach is used with a reactive routing protocol, a node has to verify whether the link to its neighbor is bidirectional before choosing it as its upstream neighbor. For this purpose, Jelger *et al.* use a protocol that involves the sending of three Neighbor Identifier (NBID) control messages.

Wakikawa *et al.* proposed a proactive and a reactive gateway discovery protocol [77]. In the proactive approach, gateways flood the network periodically with Gateway Advertisement Messages, referred to as GWADV, integrated into a Neighbor Discovery Protocol (NDP) or a proactive routing protocol. Every ad hoc node creates a route to the gateway by simply reversing the path received in the GWADV message. This reversed route is used by the ad hoc node whenever it needs to send traffic through the gateway. Bechler *et al.* proposed a passive discovery approach for internet gateways in VANets (*DRIVE* [11]). They divide the service provider into two distributed units. The first unit is located in the gateway. It permits the periodic advertisement of the gateway through the geocast transmission of messages. The second unit is located in the vehicle. When a vehicle is inside the transmission range of the gateway, it receives the advertisement message and registers the gateway information in its local database. If the client in a vehicle needs to find a gateway it checks its local database to infer whether or not a gateway is available. The overhead in *DRIVE* depends on the number of gateways. Bechler *et al.* in [11] claim that their protocol is scalable because the number of gateways is usually less than the number of vehicles. Many gateways can be present at the same time; thus, *DRIVE* uses a fuzzy-approach based on various network characteristics and application needs in order to permit the selection of the most appropriate gateway.

Wakikawa *et al.* have proposed a reactive gateway discovery protocol. It is an on-demand protocol in which the gateway does not flood the network with gateway advertisement messages. However, whenever an ad hoc node wants to find a route to the gateway, it floods a gateway solicitation message (GWSOL). When a gateway receives the GWSOL message, it sends (using unicast) a gateway advertisement message (GWADV) to the originator of the discovery request. In the GWADV message, the route to the source node is established by reversing routes received in the GWSOL messages.

Ruiz *et al.* proposed a hybrid adaptive protocol [70], that consists of selecting the value of the TTL of gateway advertisements based on the number of hops between the gateway and the source nodes. The main purpose of this approach is to reduce control overhead due to the reactive discovery scheme when there are many source nodes in the MANet. This approach also aims to reduce the overhead of the proactive protocols when the number of gateways increases. Two different protocols have been proposed: maximal source coverage and maximal benefit coverage. In [68], Ratanchandani *et al.* proposed a hybrid gateway discovery scheme. Their protocol uses concepts from Mobile IP. In their approach, foreign agents (FA) send advertisement messages to their neighboring nodes in a proactive way; however, nodes located far away from the FA operate reactively whenever they need to discover a gateway. The Time to Live (TTL) field of the IP header is set to a fixed value to limit the propagation of gateway advertisement messages. In this approach, the determination of the best TTL is therefore not straightforward, because the network conditions change frequently.

12. ***Flooding-Based Service Search***: Wewetzer *et al.* [78] provided a service discovery protocol based on flooding and they compared it with two other on-demand service discovery protocols based on different search methods in an organized VANet. They concluded that a flooding-based search method outperforms other search methods in terms of latency at the expenses of a higher traffic load.

13. ***The Vehicular Information Transfer Protocol:*** Dikaiakos *et al.* provided in [25] location-aware services over vehicular ad hoc networks using car-to-car communication. Their protocol provides time-sensitive information about the traffic conditions and the available services on the roadside. They designed the Vehicular Information Transfer Protocol (VITP) for distributed services over a vehicular network. Their protocol is an application layer protocol and is based on the location awareness concept. It is based on VITP peers. Every vehicle has a computing device that runs the VITP peer. These latter can access the vehicle's sensor and fetch valuable information. They use on-demand dynamic groups in order to collect and communicate the information from different vehicles. The collected information will be used to answer incoming requests from vehicles. The purpose of Dikaiakos *et al.* in [25] is to use the capabilities of the vehicular network to design a vehicular infrastructure that allows vehicles request location-sensitive services and get appropriate replies to their requests. The building blocks of their proposed infrastructure are: *(i)* The Vehicular Information Transfer Protocol (VITP), *(ii)* The VITP peer, *(iii)* A location encoding scheme, and *(iv)* some performance optimization protocols. The proposed VITP protocol supports both pull-based and push-based types of information dissemination. The pull-based approach is initiated on-demand and is used mainly for infotainment types of services. The push-based information dissemination is related more to safety services triggered in emergency cases. The service provision protocol provided by Dikaiakos *et al.* is based on an extended client server computing model. Vehicles' requests are location-aware. They target specific road segments. In order to answer a specific request or query, VITP peers in the specified road segment form a Virtual Ad-Hoc Server (VAHS) on-the-fly to resolve the query. Thus, a VITP transaction consists of the following phases: the first one is a query phase. In this phase, the request is forwarded through intermediate vehicles to the targeted zone using a geographic routing. The intermediate vehicles could be non VITP-enabled. When the query reaches its targeted zone and is received by a VITP

peer, the second phase, the computation phase, starts. In this phase, a VAHS is formed by the VITP peers to resolve the query. When a VAHS peer detects a predefined *Return Condition*, it generates a reply and sends it through the VANet to the source road segment, from which the query was initiated. This phase is the reply phase. When the reply reaches the source road segment and then the source VITP peer, VITP enters the last phase, which is the reply delivery phase. The *Return Condition* is defined in the query specifications and is handled by VAHS peers. The VITP protocol relies on the underlying networking protocols for the routing and transport of VITP messages. The networking in VITP is done either through the vehicular network or through other networks such as GSM/GPRS. VITP messages in the query and reply phases are routed geographically to their target location. However, VITP messages in the reply delivery phase are broadcasted in the original road segment to reach the original VITP peer. In the computation phase, the routing depends on the semantic specified in the requester VITP peer. VITP supports caching in intermediate peers and permits to peers to send replies if the *Return Condition* is satisfied. VITP guarantees the correctness of a VAHS computation by generating a unique identifier to each new request. This unique identifier is used in order to prevent from processing the same request multiple times and in order to match the requests to their corresponding replies. The identifiers are generated in such a way that insures driver's privacy. In the push-based message dissemination for alerts, VITP uses special types of messages with predetermined information. These messages will be treated as reply messages. Dikaiakos *et al.* claim that VITP can support easily *persistent queries* with simple extensions. VITP does not have built-in security features, but Dikaiakos *et al.* state that the security can be handled by other protocol layers and mechanisms. They used a Highway traffic scenario and a City traffic scenario in order to evaluate their protocol.

14. ***Geographic hash-table-based service search***: Wewetzer *et al.* provided an on-demand service discovery protocol [78], using a search based on geographic hash table (GHT). In

this protocol, the metropolitan area in subdivided into different distinct areas. A vehicle having a service uses the description of the service as an input to the hash function. The result of the function would be the location of the area where the service should be registered. The service provider registers its service and location with the resulting area. A service requester that needs a service, uses the hash function and the description of the service to find the area where the service is registered. Then it sends the request to the resulting area. A vehicle in the resulting area knows the location of the service and sends the request to the service provider. This latter replies directly to the service requester.

15. ***Distributed hash-table-based service search***: Wewetzer *et al.* provided an on-demand service discovery protocol [78], using a search based on distributed hash table (DHT). The distributed hash table is more generic than the geographic hash table. Space identification in GHT is geographic, whereas in DHT the space identification is virtual. In DHT, each vehicle acquires a virtual ID when it joins a DHT and maintains a neighbor table in the virtual ID space. Each vehicle forwards the request to another vehicle closer in the ID space than itself. Such that, the request for a certain key will reach certainly the responsible vehicle. Similar to GHT, in DHT a vehicle registers its service using a hash function and a requester requests a service using also the hash function.

16. ***Context-Aware Migratory Services in Ad Hoc Networks:*** The main purpose of context-aware service discovery protocols is to ensure that service providers continue to guarantee the required quality of service when there is a change in the client context or the service context. For example, let us consider a service that consists of monitoring a certain region, if the vehicle providing the service moves away from the required region, then its quality of service decreases or it will not be anymore suitable to provide the service. Due to this context modification, it is better that another vehicle if available in the required context takes over in order to provide the same service with better quality of service. In the following paragraph, we describe some context-aware discovery mecha-

nisms. In [69], Riva *et al.* provided a context-aware service discovery protocol. In their protocol, when a vehicle needs a service it requests vehicles in the region of interest. Since vehicles in the region of interest are moving with different speeds, services can migrate from one vehicle to another vehicle in order to be able to accomplish their tasks. In fact, in their protocol, when a vehicle is not able anymore to satisfy a user request, the service finds another suitable node and migrates to this node in a transparent way to the user. A small delay will be incurred but the service will not be interrupted. The physical location of the service can be in many node, but the client will interact with only one virtual node. As a results, a client-service interaction is guaranteed.

17. ***Multi Hop Wireless Ad Hoc Parking Meter Network:*** Basu *et al.* proposed a multi hop wireless ad hoc parking meter network (PMNET) [10], that permits to discover free parking spaces. Two main components compose the PMNET: *(i)* parking meters which are static nodes, and *(ii)* vehicles which are mobile nodes. Non-volatile information of a parking meter, such as its geographical coordinates and its street location, are stored permanently in the parking meter memory. The availability of a spot, its fee and its time limit are considered as volatile information. A query received by a parking meter is handled by its *query processing module*. If a parking meter can satisfy the query, then a response message is unicast to the requesting vehicle. Otherwise, the parking meter rebroadcasts the query to its neighboring static nodes. The *status update processing module* updates the volatile data in each parking meter. In order to reduce the bandwidth usage and the latency, Basu *et al.* used a hybrid approach for query and status updates. Parking meters having the same street name, form a cluster. A clusterhead could be either identified during the installation of parking meters, or it could be elected by all parking meters in the same cluster using a leader election mechanism. In PMNET, clusterhead nodes form an overlay and disseminate status updates through unicast whenever an update is reported to a clusterhead from its cluster member. Thus, a vehicle needs to query a nearby clusterhead parking meter in order to get its response. Communication

between parking meters is done through unicast. Communication between vehicles or parking meter to vehicle is done through a cluster based MANet routing protocol.

18. ***Virtual Mobile Nodes for Mobile Ad Hoc Networks:*** Dolev *et al.* proposed a service discovery mechanism based on virtual mobile nodes for mobile ad hoc networks [26]. They stated that their approach could be applied to vehicular networks. They used two types of nodes: *(i)* physical nodes or real nodes and *(ii)* virtual mobile nodes (VMN). In a vehicular network, the real nodes are the cars within a predetermined region and the virtual nodes are agents moving in a predictable, predetermined path. Virtual nodes collect and aggregate data regionally and temporally in a predefined region. A vehicle sends a request to a VMN and specifies the required data to be collected and its way of aggregation. VMNs return the necessary data to the client.

19. ***Context Aware Session Management for Services in Ad Hoc Networks:*** Handorean *et al.* proposed a context aware session management for services in ad hoc networks [36]. In their protocol, a session is defined as a lasting connection between a client and a service, as opposed to a lasting connection between a client and a server in traditional sessions. The server implementing the service may be a mobile code migrating from host to host to follow the client as it moves in the space. That is why Handorean *et al.* called their protocol *"Follow me"*. The protocol is able to reconnect to another service provider transparently to the client, if the new service provider is considered to be better. In [36], Handorean *et al.* assume the definition of better as the time to disconnection between the two hosts.

20. ***A hybrid approach for location-based service discovery in vehicular ad hoc networks:*** Klimin *et al.* proposed a hybrid service discovery protocol in ad hoc vehicular networks [45]. Their approach combines the proactive dissemination of advertisement messages and the reactive propagation of discovery requests. Their protocol is based on the geocast addressing of control messages. They state that through this approach,

the message overhead is decreased, thus the bandwidth is increased. For their experimentation, they provided a testbed.

2.4 Classification of Service Discovery Protocols

In the literature, different discovery modes have been investigated: (i) Push-based discovery [18, 29, 58, 79, 80, 13, 83, 11], where the provider disseminates service information in all the network, (ii) Pull-based discovery [69, 78, 25], where the requester sends a request to find a specific service, and (iii) Hybrid-based discovery [10, 26, 36, 45] where the push and the pull modes are used at the same time. In the following, we provide a classification of existing service discovery protocols in VANets according to their discovery mode.

2.4.1 Push-based Service Discovery Mode

In a push-based discovery approach, the service provider does not wait for service requests from requesting vehicles. Instead, it advertises its service in a proactive way and disseminates the service information to all the vehicles in the vehicular network. Many protocols in this category have been proposed in vehicular networks. There are two basic tunable parameters in the proactive mode that should be handled carefully for efficient proactive service discovery protocols: (i) the advertisement range of service providers, and most importantly (ii) the advertisement frequency of service providers. In vehicular networks, these parameters depend mainly from the mobility rate of service providers (high, medium, low), the type of application (road safety and security, traffic monitoring and driving comfort), network characteristics (congestion, failure, collisions), service availability, etc.

Another aspect that affects proactive service discovery protocols in VANets is the way the advertised information is aggregated and disseminated. We depict four different structures: (*i) Quad tree structure* [18], which permits the aggregation and dissemination of information in hierarchical way such that four disjoint areas of a lower level cover one area of the

Table 2.1: Push-based discovery protocols comparison: Road segment structure-based protocols

characteristics /protocols	Warning Delivery [29]	VITP [25]	TrafficView [58]	SOTIS [79]
Architecture	Decentralized	Decentralized	Decentralized	Decentralized
Operation mode	Push	Pull/Push	Push	Push
Structure	Flat	Flat	Flat	Flat
Road structure	Road segment	Road segment	Road segment	Road segment
Routing	Broadcast	Geocast / broadcast / GSM/GPRS	Broadcast	Broadcast
Medium access protocol	CSMA	802.11	IEEE 802.11b	TDMA
Traffic model	highway	highway/urban	highway	highway
Traffic Generator	not mentioned	SUMO[]	VISSIM [58]	cellular automaton approach [59]
Data Prioritization	No	No	Through aggregation	considered for emergency and through data analysis
Delay tolerant	No	Yes	No	Yes
Security/Privacy	No	No	No	No
Internet connectivity	No	No	No	No
Data Type	Homogenous	Homogenous	Homogenous	Homogenous
Fidelity	not considered	not considered	not considered	not considered
Information accuracy	High	Average	Average	High
Faireness/QoS	No	No	No	Yes
Location Aware	Yes	Yes	Yes	Yes
Cache size	not mentioned	not mentioned	unlimited	not mentioned
Deleting process	not considered	not considered	not considered	not considered
Main overhead type	broadcast	Queries routing	Information diffusion	Periodic report broadcast
Message overhead	Medium	Medium	medium	medium
Latency	low	Medium	Medium	Medium
Success Rate	High	Medium	Medium	High
Dropping rate	Not considered	Average	Not considered	Not considered
Scalability	Yes	Yes	Yes	Yes
Network layer	Application layer	Application layer	Application layer	Application layer
Mobility support	event-driven/ range control	not required	Periodical update	event-driven/ range control
Mobility handling	high	low/medium/high	low/medium	low/medium/high
Delay sensitivity	yes	yes	yes	yes
Service description	predefined attributes	predefined attributes	predefined attributes	predefined attributes
Bootstrapping	Broadcast	geographic	Broadcast	Broadcast
Search method	directory-less	directory-less	directory-less	directory-less
Discovery scope	spatial distribution	location-based	spatial distribution	spatial distribution
Service information update	notification	polling	notification	notification
Service selection	Not considered	manual	manual	Not considered
Service invocation and access	Network location	network location	network location	network location
Security	Not considered	Not built-in	Not considered	Not considered

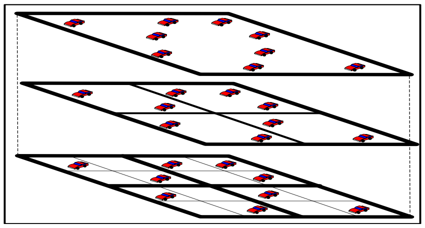
higher level as illustrated in Figure 2.2(a); *(ii) Road segment structure* [29, 25, 58, 79], which permits the aggregation and dissemination of information by dividing the road in consecutive disjoint segments as illustrated in Figure 2.2(c); *(iii) Grid structure* [80], where the vehicular network is organized in a grid for information dissemination as illustrated in Figure 2.2(b); and *(iv) Line road structure* [13, 83] where information dissemination follows a predeter-

Table 2.2: Push-based discovery protocols comparison: Quad tree, grid and road line structure based protocols

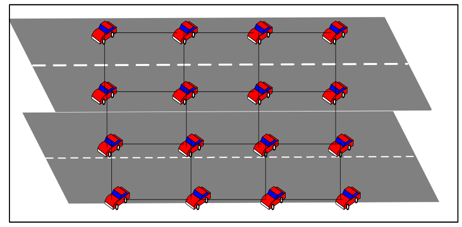
characteristics /protocols	Free Parking [18]	PPB[80]	CCA [13]	VCWC [83]
Architecture	Decentralized	Decentralized	Decentralized	Decentralized
Operation mode	Push	Push	Push	Push
Structure	Clustered and hierarchical	Flat	Flat	Flat
Road structure	Quad Tree	Grid	Road line	Road line
Routing	Broadcast	Broadcast	Broadcast	Broadcast
Medium access protocol	IEEE 802.11	802.11b	802.11	802.11
Traffic model	city	city	highway	highway
Traffic Generator	VISSIM [58]	their own in grid network	their own	their own
Data Prioritization	Through the selection strategy	No	Yes	yes through differentiation
Delay tolerant	Yes	No	Yes	Yes
Security/Privacy	No	No	No	No
Internet connectivity	No	No	No	No
Data Type	Homogenous	Homogenous	Homogenous	Homogenous
Fidelity	not considered	not considered	not considered	not considered
Information accuracy	High	Average	High	High
Faireness/QoS	No	No	Yes	Yes
Location Aware	Yes	Yes	Yes	Yes
Cache size	unlimited	not mentioned	not mentioned	not mentioned
Deleting process	not considered	not considered	not considered	not considered
Main overhead type	Information dissemination	Broadcast	Broadcast	Broadcast
Message overhead	Medium (controlled)	medium	medium	Average
Latency	Low	High	Low	Low
Success Rate	High	Medium	High	High
Dropping rate	Not considered	Not considered	Low	Not considered
Scalability	Yes	No	Yes	Yes
Network layer	application layer	cross-layer	cross-layer	cross-layer
Mobility support	periodic update	periodic update	event-driven update	event-driven update and transmission frequency control
Mobility handling	low/medium	low/medium	medium/high	medium/high
Delay sensitivity	No	No	Yes	Yes
Service description	attributes/names	attributes/names	attributes/names	attributes/names
Bootstrapping	broadcast	broadcast	broadcast	broadcast
Search method	directory-less	directory-less	directory-less	directory-less
Discovery scope	spatial-distribution	spatial-distribution	spatial-distribution	spatial-distribution
Service information update	notification	notification	notification	notification
Service selection	Manual	Manual	Automatic	Automatic
Service invocation and access	network location	network location	network location	network location
Security	not considered	not considered	not considered	not considered

mined road line as illustrated in Figure 2.2(d). Tables 2.1 and 2.2 present the characteristics and features of the push-based service discovery protocols.

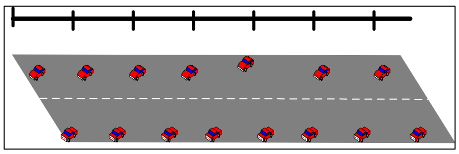
Biswas *et al.*[13] proposed a proactive dissemination protocol for safety applications that follows the *line road structure*. A vehicle that detects an emergency starts broadcasting warning messages periodically. In a first version of their protocol, every intermediate vehicle that



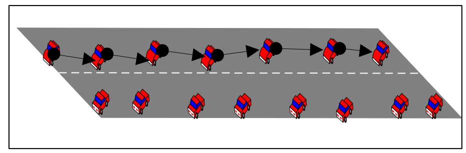
(a) Quad tree structure-based service discovery



(b) Grid structure-based service discovery



(c) Segment road structure-based service discovery



(d) Line road structure-based service discovery

Figure 2.2: Classification of push-based service discovery protocols in VANets

receives the warning message from a preceding vehicle rebroadcasts the message. In a second version, an intermediate vehicle that has broadcasted the message and then receives it again from a succeeding vehicle stops the retransmission of the message in order to reduce message collision. Yang *et al.* [83] also proposed a proactive dissemination protocol for safety applications where information dissemination and aggregation follow the *line road structure*. Yang *et al.* proposed a transmission rate decreasing algorithm for congestion control. They proposed as well a state transition mechanism to eliminate warning messages redundancy. The transmission range of emergency messages is limited to a predetermined safety zone.

Fracchia *et al.* [29] proposed a proactive warning dissemination protocol for safety type of applications where information aggregation and dissemination follows a *road segment structure*. They limited the dissemination range to a safety zone. The vehicle starts new broadcasts of the same warning message every predefined period and intermediate vehicles decide to rebroadcast the message with a probability α. Another protocol that follows the same *road structure* for information aggregation and dissemination was proposed in [25]. The protocol is basically reactive, however, in case of emergency, the protocol performs in a proactive way and disseminates emergency messages.

Wischoff *et al.* [79] proposed a proactive dissemination protocol using the *road segment structure*. The protocol permits to send two types of reports: Periodic report that contain traffic information, and Emergency reports that are sent in case of emergency. Nadeem *et al.* [58] proposed a proactive traffic information dissemination protocol for vehicular networks using the *road segment structure* as well. In this protocol, every vehicle broadcasts traffic information about itself and its neighbors periodically.

Caliskan et al. [18] proposed a proactive service discovery approach for spatio-temporal type of applications using the *Quad tree structure*. In their protocol, information about service providers is broadcasted every predefined periodic interval by considering the spatio-temporal characteristics of the places. Their protocol generates two types of reports about service providers information: the first type is for atomic information and the second report is for aggregate information. The range of advertisement in their protocol depends on the type of report. Atomic information is distributed in local proximity and aggregate information is distributed in wider area.

Wolfson *et al.* [80] proposed a proactive physical resource discovery protocol in VANets using the *Grid structure*. In this protocol, every vehicle broadcasts the M most relevant resource information to its neighbors periodically. Bechler *et al.* [11] proposed a proactive gateway discovery protocol for vehicular networks. Gateways advertise themselves periodically in the vehicular network.

In a push-based discovery approach, the service provider does not wait for service requests from requesting vehicles. Instead, it advertises its service in a proactive way and disseminates the service information to all the vehicles in the VANet. This approach achieves good connectivity but results in a high overhead and consequently protocols in this category have limited scalability, unless an efficient aggregation technique is used. Quad tree structure-based service discovery protocols suffer from a very high redundancy on the disseminated information; this high redundancy increases the loss of accuracy of the services provided. In addition, quad tree structure-based protocols are limited to fixed services; which means that the location of the service provider is fixed all over the time. Moreover, the static service providers have to be organized in such a way that permits the feasibility of the quad tree overlay.

Road segment structure-based service discovery protocols are mainly designed for the warning delivery and traffic monitoring types of services. In fact, in this type of protocols, information is aggregated according to the disjoint road segments; which permits to neighboring vehicles to be aware of the current traffic conditions and close dangers. Grid structure-based service discovery protocols suffer from a high latency and thus they cannot be applied to delay sensitive types of applications. Road line structure-based service discovery protocols are applied to delay-sensitive type of applications; which are mostly related to safety applications. We believe that the proactive dissemination of service information is required for safety types of applications. However, knowing the drawbacks of the proactive approach, aggregation techniques have to be used. Unfortunately, the aggregation of information can decrease the level of accuracy on the distributed information. Thus, an open issue in proactive discovery protocols is how to permit the aggregation of information while preserving its level of accuracy.

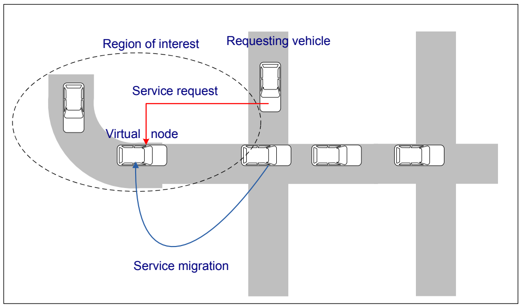
2.4.2 Pull-based Service Discovery Mode

In the reactive mode, service providers do not advertise their services, they wait for service requests from requesting vehicles. If a driver needs to discover a service, he sends a service request in the vehicular network. Vehicles that have the service that satisfies the request, send a service reply to the requesting vehicle. A simple approach adopted by requesters for reactive service discovery consists of *flooding* the whole network with request messages such as the discovery protocol presented in [78] for unorganized VANets. This approach is naive and not promising, because it floods the whole network with unnecessary request messages, mainly if the number of requesters is large and the density of the VANet is high. In order to avoid flooding the whole network with request messages, other techniques have been adopted by service requesters in VANets. These techniques consists of (i) *forwarding requests to a selected set of neighbors*; or (ii) *use of context awareness concept for requests propagation*.

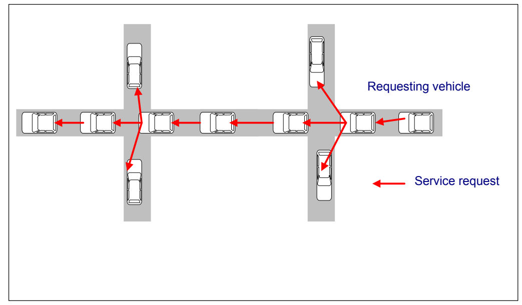
Table 2.3: Pull-based discovery protocols comparison

characteristics/protocols	Migratory Services [69]	flooding [78]	GHT and DHT [78]
Architecture	Decentralized	Decentralized	Decentralized
Operation mode	Pull	Pull	Pull
Structure	Flat	Flat/Hierarchical	Flat/Hierarchical
Road structure	Road line	Square	Square
Routing	Geographical/ Content-based on-demand routing	Broadcast	Overlay routing
Medium access protocol	IEEE 802.11b	IEEE 802.11	IEEE 802.11
Traffic model	highway/city	city	city
Traffic Generator	Micro-VTG [58][84]	VISSIM [52]	VISSIM [52]
Data Prioritization	No	No	No
Delay tolerant	No	Yes	Yes
Security/Privacy	Yes	No	No
Internet connectivity	No	No	No
Data Type	Homogenous	Homogenous	Homogenous
Fidelity	not considered	not considered	not considered
Information accuracy	High	High	High
Faireness/QoS	No	No	No
Location Aware	Yes	Yes	Yes
Cache size	not mentioned	not mentioned	not mentioned
Deleting process	not considered	not considered	not considered
Main overhead type	Message propagation	Message propagation	Message propagation
Message overhead	Average	High	Average
Latency	Medium	Low	Medium
Success Rate	High	High	High
Dropping rate	Not considered	Not considered	Not considered
Scalability	Yes	No	Yes
Network layer	cross layer	Application-layer	Application-layer
Mobility support	migration	not required	periodical update
Mobility handling	low/medium	low/medium	low/medium
Delay sensitivity	No	No	No
Service description	attribute-names	attribute-names	attribute-names
Bootstrapping	geographical routing	broadcast	computation-based
Search method	directory-less	directory-less	directory-based
Discovery scope	context-based	spatial-distribution	location-based
Service information update	notification	not considered	notification
Service selection	automatic	not mentioned	not mentioned
Service invocation and access	communication mechanism	network location	network location
Security	Yes	No	No

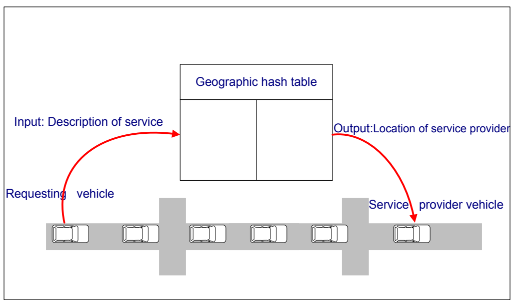
Dikaiakos *et al.* [25] proposed a location-based reactive service discovery protocol for vehicular networks. In order to reduce the congestion in the VANet, requests are propagated to selected neighbors using geographic routing until they reach the region of interest specified



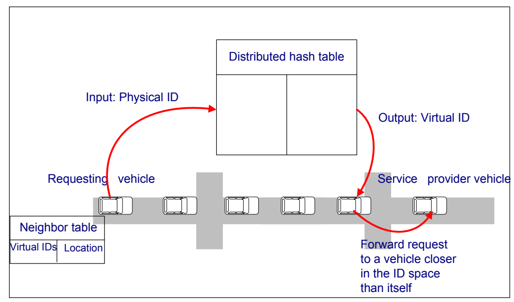
(a) Context-aware discovery



(b) Flooding-based discovery



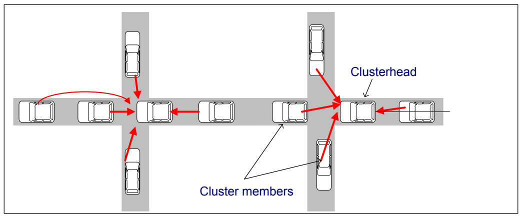
(c) Geographic hash table-based discovery



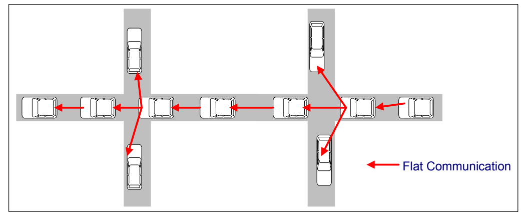
(d) Distributed hash table-based discovery

Figure 2.3: Classification of pull-based service discovery protocols in VANets

in the driver's request. Riva *et al.* [69] proposed a context-aware service discovery protocol for vehicular networks. In their protocol, a driver interacts with only one virtual node. The physical location of the virtual node can be in many nodes. In pull-based service discovery approach, service providers do not advertise their services, they wait for service requests from requesting vehicles. In this approach, if the number of service requesters is high, then the number of messages exchanged for the discovery of the service is also very high. Consequently, the scalability is not guaranteed. Riva *et al.* [69] used a migratory service approach, it is clear that this approach is dedicated to context-aware type of service. Other techniques can be used in order to limit the propagation of request messages such as: (i) limit the time-to-live (TTL) of request messages; (ii) or increment gradually the number of hops of request messages. Table 2.3 presents the characteristics and features of the pull-based service discovery protocols.



(a) Clustered structure-based service discovery protocols



(b) Flat structure-based service discovery protocols

Figure 2.4: Classification of hybrid service discovery protocols in VANets

2.4.3 Hybrid Service Discovery Mode

In the hybrid service discovery mode, both proactive and reactive techniques collaborate for the service discovery. We distinguish Clustered structure-based service discovery protocols (illustrated in Figure2.4(a)) and Flat structure-based service discovery protocols (illustrated in Figure 2.4(b)). In the *clustered structure-based* service discovery category, nodes are organized in clusters. In each one, there is a clusterhead that handles the information of its members and can communicate with neighboring clusterheads. Cluster members send their requests to their clusterhead. This latter, retrieves the required information from its cluster members or from its neighboring clusterheads and replies back to the requesting node. The protocol proposed in [41] considers a central portal that collects data from providers and processes service requests. In [21] a three layer architecture is considered. The higher layer comprises ethernet connected servers that handle service information storage and requests.

In [5], registries and mediators are used for hybrid service discovery. In the protocol proposed in [10] clusterhead nodes form an overlay and store resources information. A requester sends its requests to its nearest clusterhead. If this latter has the required resource information, it sends back a service reply to the requester. Otherwise, the clusterhead sends the service request to its neighboring clusterheads to look for the requested service. In [26], Dolev *et al.* proposed a cluster-based hybrid discovery approach based on virtual mobile nodes acting as agents. These latter, collect and aggregate data regionally and temporally in a predefined region. They also receive service requests from vehicles, then collect and aggregate the required

Table 2.4: Hybrid discovery protocols comparison

characteristics /protocols	PMNET [10]	VMN [26]	Follow me [36]	Hybrid-SD [45]
Architecture	Decentralized	Decentralized	Decentralized	Decentralized
Operation mode	Hybrid	Hybrid	Hybrid	Hybrid
Structure	Clustered	Clustered	Flat	Flat
Road structure	Grid	Not specified	Not specified	Not specified
Routing	Unicast	Broadcast	Broadcast	Geocast
Medium access protocol	i-Bean	Not specified	802.11	Not specified
Traffic model	city	highway/city	Not specified	Not specified
Traffic Generator	not used	Not specified	Not specified	Not specified
Data Prioritization	No	No	No	No
Delay tolerant	Yes	Yes	Yes	Yes
Security/Privacy	No	No	No	No
Internet connectivity	No	No	No	No
Data Type	Homogenous	Homogenous	Homogenous	Homogenous
Fidelity	not considered	not considered	not considered	not considered
Information accuracy	Average	Average	High	High
Faireness/QoS	No	No	No	No
Location Aware	Yes	Yes	Yes	Yes
Cache size	not mentioned	not mentioned	not mentioned	not mentioned
Deleting process	not considered	not considered	not considered	not considered
Main overhead type	Unicast	Broadcast	Broadcast	Message propagation and dissemination
Message overhead	Average	not studied	not studied	
Latency	Low	Medium	not studied	not studied
Success Rate	Medium	Medium	not studied	High
Dropping rate	Not considered	Not considered	Not considered	Not considered
Scalability	Yes	Yes	Not considered	Not considered
Network layer	application-layer	application-layer	application-layer	cross-layer
Mobility support	event-driven update	periodical update	event-driven update	periodical update
Mobility handling	low/medium	low/medium	low/medium	low/medium/high
Delay sensitivity	No	No	No	No
Service description	attribute-name	attribute-name	attribute-name	attribute-name
Bootstrapping	geographic dissemination	broadcast	broadcast	geographic dissemination
Search method	directory-based	directory-based	directory-less	directory-less
Discovery scope	location-based	location-based	context-based	location-based
Service information update	notification	notification	not mentioned	notification
Service selection	manual	not mentioned	manual	manual
Service invocation and access	network location	not mentioned	network location	network location
Security	No	No	No	No

data to be returned to requesters.

In the *flat structure-based* service discovery protocols category, nodes in the vehicular network communicate with each other in a flat way; which means that there is no level of aggregation of the information. In [45], Klimin *et al.* proposed a flat structure-based approach for hybrid service discovery. Service providers disseminate advertisement messages in a pre-defined number of hops and service requesters discover the services in a proactive way if they are inside the advertisement zone of the requested provider. Otherwise requests are transmitted reactively until they reach the advertisement zone of the service provider. In [78], Wewetzer *et al.* proposed a flat-structure-based hybrid approach for service discovery in VANets. They use geographic and distributed hash tables for service location computation.

In this approach, protocols combine both characteristics of the push-based approach and the pull-based approach. Protocols presented in this section prove that the cluster-based structure is more efficient and more scalable than the flat structure. Tables 2.4 presents the characteristics and features of the hybrid-based service discovery protocols.

2.4.4 Comparison and Discussion

We notice from the previously discussed protocols that safety applications use the proactive mode for service discovery and dissemination. This is because these applications are mainly event-driven and are prioritized over other applications. For the other types of applications, the choice of the best discovery mode depends from many factors. For example, if the number of service providers is considerably lower than the number of service requesters then a proactive discovery mode outperforms the reactive discovery mode in terms of latency and overhead. However, if the number of service requesters is lower than the number of service providers, then a reactive discovery mode is better than a proactive discovery mode. A hybrid discovery mode can be used in order to provide reasonable performance results. A basic challenge in the hybrid mode is about how to determine the range and the frequency of service providers advertisement. In fact, a large range and frequency of the advertisement zone incurs the same drawbacks of the proactive mode, and a reduced range and frequency of advertisement incurs the same disadvantages of the reactive mode. Consequently, these two factors should be chosen very carefully in the hybrid discovery mode. The best way to deal with this challenge is to investigate adaptation mechanisms of the range and frequency of advertisement depending from the network characteristics (traffic pattern, mobility, density, etc) and conditions (congestion, contention, etc).

2.5 Summary

In this chapter, we have reviewed the basic subjects related to this thesis. First, we started by providing an overview about VANets, their characteristics, and their main applications. Then, we discussed existing service advertisement and discovery protocols in VANets and we classified them according to their discovery mode. We summarized the main characteristics and features of the reviewed service discovery protocols in Tables 2.1, 2.2, 2.3, and 2.4.

Chapter 3

Gateway Discovery Protocol: LAGAD

3.1 Introduction

Internet access from vehicular networks is gaining great interest from the research community. In fact, vehicles should be able to connect to the Internet and communicate with heterogeneous networks through gateways. Guaranteeing safety on the road is the main objective of vehicular networks. Safety applications need to collaborate with other types of services for efficient safety assurance. Consequently, many services would coexist with safety applications, including the gateway discovery service, and share a limited bandwidth. Any solution to the gateway discovery problem in vehicular ad hoc networks is subject to this limitation. In the following, we discuss our motivation and research challenges, then we state our contribution and objectives, and finally we present the chapter organization.

3.1.1 Motivation and Research Challenges

A gateway discovery mechanism is needed to allow drivers and passengers seeking Internet access to discover gateways and connect to the Internet. The gateway discovery mechanism needs to work without any prior configuration; i.e., if vehicles are circulating on the roads forming a vehicular network, they must not be required to configure the gateway discovery mechanism before they start to discover gateways in the vehicular network. A driver's or passenger's request to discover a gateway is handled at the nearest gateway. A reply message is then sent to the gateway requester. Many gateway discovery protocols have been proposed in the literature [42, 77, 70, 68, 31]. Basically, they are classified into three categories: (1) proactive approach [42, 11], where gateways advertise themselves in the whole network; (2) reactive approach [77] where requesters need to flood the whole network for gateway discovery; and (3) hybrid approach [70, 68], where gateways advertise themselves in a predetermined number of hops and requesters send their gateway requests up to the advertisement zone.

3.1.2 Contribution and Objectives

In this chapter, we present a novel hybrid adaptive Location-Aided Gateway Advertisement and Discovery (LAGAD) protocol for VANets. Our protocol permits gateway clients to continue being updated with the route towards the gateway. In other words, given a set of gateways and assuming that each road component is aware of its position, the goal of this work is to let each gateway requester car discover nearby gateways and gain sufficient information to route packets towards the closest gateway while guaranteeing the network scalability. LAGAD protocol has the following unique and efficient characteristics for gateway discovery in VANets. First, it is built on top of the network layer. Second, it uses channel diversity. Third, it is based upon a location-aided adaptation of the advertisement zone of the gateway. Unlike previous work, our protocol has many characteristics that improve its efficiency. In fact, some of the previous work required a prior configuration, where each car needed to download the position of the gateways from the Internet and perform position-based routing towards them. The car would then be able to update on the status of the gateways every once in a while when it is connected to one of the gateways. However, our protocol does not require any prior configuration and can perform efficiently in ad hoc manner.

Existing gateway discovery protocols without prior configuration can be classified into three categories: proactive, reactive or hybrid discovery protocols. The proactive approach achieves good connectivity but results in high overhead if the number of gateways is high. The reactive approach incurs low overhead for small networks but suffers from poor scalability. The hybrid approach is supposed to be more efficient than proactive and reactive approaches; however, determining the radius of the advertisement zone is the main concern. A small advertisement zone could result in a high reactive overhead, and a large advertisement zone could result in a high proactive overhead. The overhead is related to the number of control messages forwarded by each road component. Our protocol provides a tradeoff between the reactive and the proactive gateway discovery solutions and bypasses the drawbacks of previous hybrid protocols, which consists of adjusting the scope for the gateway advertisement. It also overcomes the problem of dynamically determining the scope of the proactive zone when using the maximal source coverage concept, which can lead to unnecessary flooding of the network in many scenarios. Our mechanism is not only hybrid, but is also adaptive because the determination of the gateway's advertisement zone adapts to the changing number and location of gateway clients, as well as the number of existing gateways which bypasses the drawbacks of existing hybrid approaches. Moreover, most of the existing gateway discovery protocols are based on the application layer. Thus, there is a need to discover a gateway and routing information to the gateway through routing messages.

Our proposed protocol is network-layer-based, i.e, it benefits from the routing information to find the gateway service and the routing information to the gateway at the same time saving the overall bandwidth. Finally, our protocol uses diverse channels to exchange discovery and routing packets decreasing the congestion on single channels and decreasing the delay of gateway discovery. In fact, the interference between multiple simultaneous transmissions in wireless ad hoc networks, results in the reduction in their capacity [32]. In order to improve the capacity of these networks, equipping every node with multiple radios is considered as a promising solution [7, 8]. In fact, as opposed to single radio where the capacity of relay nodes is halved, multiple radios permit each node to transmit and receive at the same time. Moreover, using two radios permits a node to transmit on two different channels at the same time which enlarges its usage of the radio spectrum. In addition, radios can use different frequency bands that have different characteristics (bandwidth, range, etc.) which improve the network performance. Finally, 802.11 radios are becoming inexpensive; thus they can be acquired easily. The main objective of our proposed protocol is to provide a high success rate for gateway queries, while guaranteeing a low response time and network scalability.

3.1.3 Chapter Organization

The remainder of this chapter is organized as follows. Section 3.2 contains a detailed description of our gateway discovery algorithm. Section 3.3 presents the proof of correctness of our protocol. Section 3.4 presents our discussion of the complexities' computation. Section 3.5 reports the simulation experiments we have conducted in order to evaluate the performance of our algorithm and we compare our protocol with existing gateway discovery approaches. Finally, Section 3.6 provides our conclusion for this chapter.

3.2 The Location-Aided Gateway Advertisement and Discovery (LAGAD) Protocol for VANets

In this section, we shall present the design of our proposed gateway discovery protocol for large-scale Vehicular Networks. Before we proceed further, let us present our system model.

3.2.1 System Model

Consider a VANet in a Manhattan model[1]. Our system comprises two types of components: Roadside Gateways (RGs), and Road Vehicles (RVs): ***(i) RGs*** could be either Roadside Router Gateways (***RRG***) or Roadside Internet Gateways (***RIG***). *RRGs* allow the integration of different access technologies and access modes; *RIGs* allow the connection and access to the Internet. ***(ii) RVs*** are the cars circulating along the roads equipped with various devices.

[1] A mobility model that uses grid road topology.

They are characterized by their high mobility and their increased density. Road Vehicles are connected to each other in an ad hoc way to form the Vehicular ad hoc network (VANet). A source vehicle is the vehicle that must find an appropriate gateway.

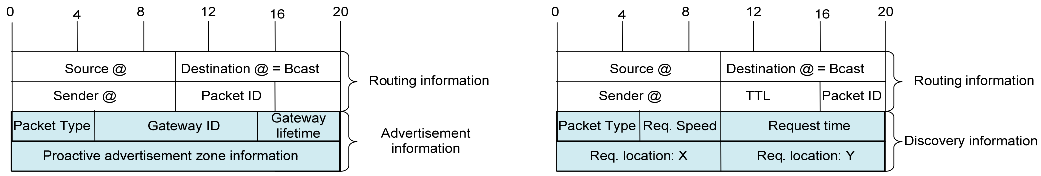
In our system model, we suppose that vehicles move on a two dimensional plane. We consider a simple Manhattan model of straight lanes and perpendicular lanes. Lanes are L meters long and l meters large. We consider a number of nG gateways located on the straight lanes. Gateways are located at the intersections of straight and perpendicular lanes. We model the arrival process to the VANet as a Poisson process with an arrival rate (λ). The density of the network is *Density* and it provides the number of vehicles per meter square in the VANet. The density of a road section is equal to the density of the VANet. We assume that a road section has a fixed length and a fixed surface. The average speed of a vehicle is v(m/s). We assume the random variable N defines the number of vehicles in a road section of length x. The probability that there are n vehicles in a road section of length (x) is given by:

$$P[N(x) = n] = \frac{(\frac{x}{S}\lambda)^n}{n!} e^{-\lambda \frac{x}{S}} \tag{3.1}$$

Before we proceed further, let us present the format of LAGAD packets and information tables.

3.2.2 LAGAD Packets and Tables Format

Figure 3.1(a) illustrates the format of GPD (Gateway Proactive Discovery) packets that we also refer to as gateway advertisement messages $GAdv_{msg}$. GPD packets contain routing information and advertisement information. The routing information consists of: *(i)* the gateway ID (source address); *(ii)* the destination address which is the broadcast; *(iii)* the sender address which is the identifier of the sending vehicle for routing purposes; and *(iv)* the packet ID in order to permit the receiver to eliminate redundant GPD packets. The advertisement information of the GPD packet consists of: *(i)* the packet type (advertisement, request, or



(a) Format of the Gateway Proactive Discovery (GPD) packet

(b) Format of the Gateway Reactive Discovery (GRD) packet

Figure 3.1: Format of the Gateway Proactive and Reactive Discovery packets

reply); (ii) the identifier of the gateway; (iii) the lifetime of the gateway; and *(iv)* the information of the location-aided advertisement zone that will permit receiving vehicles to forward or decline the GPD packet.

Figure 3.1(b) shows a GRD (Gateway Reactive Discovery) packet, that we also refer to as $GReq_{msg}$. GRD packets contain routing information and discovery information. The routing information consists of: *(i)* the gateway requester ID (source address); *(ii)* the destination address which is the broadcast used to find an appropriate gateway; *(iii)* the sender address which is the identifier of the sending vehicle for routing purposes; *(iv)* the time to live of the packet (TTL); and *(v)* the packet ID used to permit the receiver to eliminate redundant GRD packets. The discovery information of the GRD packet consists of: *(i)* the packet type (advertisement, request, or reply); and *(ii)* the location information and the speed and current time of the requesting vehicle that will be used by the eventual gateway to determine the location-aided advertisement zone. GPD packets are used by a gateway or a vehicle in the gateway's advertisement zone in order to advertise the gateway in the proactive discovery zone. Outside the proactive discovery zone, GRD packets are used to generate gateway discovery queries.

Every vehicle using our LAGAD protocol uses a gateway table and a routing table to respectively store gateways' information and routing information. The format of a gateway table is illustrated in Figure 3.2(a). It consists of *(i)* the gateway identifier and *(ii)* the gateway information. The routing table format is illustrated in Figure 3.2(b). It consists of *(i)* the identifier of the destination node; *(ii)* the identifier of the next hop node; and *(iii)* the interface number for channel diversity purposes.

Gateway ID	Gateway Information
Gateway 1	LifeTime
	Other information
Gateway 2	...
Gateway ...	...

(a) Format of the gateway information table

Destination	Next hop	Interface
Vehicle 1	Vehicle ID	Interface number
Vehicle 2	...	...
Vehicle ...	...	...

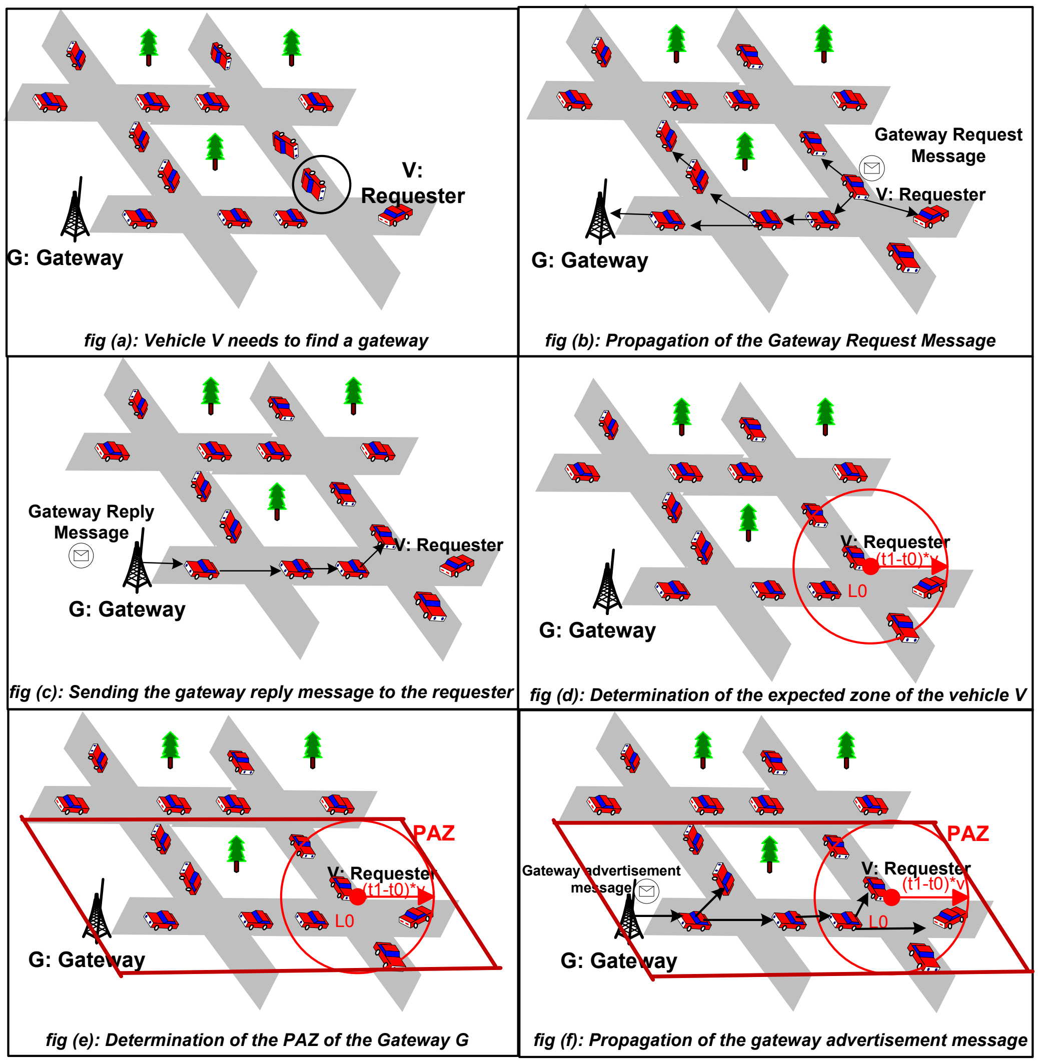
(b) Format of the routing table

Figure 3.2: Format of the gateway information and routing tables in LAGAD

3.2.3 Description of Our Proposed LAGAD Protocol

In what follows, we will describe our LAGAD protocol. In Figure 3.3, we present an illustrative example for the execution of our *LAGAD* protocol. We provide a detailed description of the main features of our protocol. For this purpose, we describe our location-aided gateways advertisement mechanism and we explain how we integrated our protocol in the network layer and how we made use of the channel diversity feature in our scheme. Our gateway discovery protocol is based on the location information of source vehicles. We have used several concepts from some well known location-based routing protocols [14, 9, 46, 20]: the Location-Aided Routing (LAR) in mobile ad hoc networks [46], the Distance Routing Effect Algorithm for Mobility (DREAM) [9] and the adaptive mesh-based protocol for Geocast Routing [20]. The main purpose of these location-based routing protocols is to determine the expected zone of a destination *D*, and to forward routing requests so that network overhead is reduced. In our protocol, we investigate the different mechanisms related to the determination of expected zones and forwarding zones. Then, we propose a location-based gateway discovery and advertisement protocol for VANets.

We call our approach Location-Aided Gateway Advertisement and Discovery (LAGAD) because it refers to vehicles' location information in order to reduce control overhead. The vehicles' location information can be provided by the Global Positioning System (GPS) based protocols [27, 3, 63] or by localization protocols [17, 16, 6, 86, 75]. In our proposed protocol, we will make use of GPS as a localization tool. Thanks to GPS, every vehicle can know its physical location information. In the real word a vehicle's GPS computed coordinates are



Execution sequence:

[*Seq*0 :] A vehicle *V* is seeking a gateway.

[*Seq*1 :] A gateway request is generated by *V* and is propagated to the gateway.

[*Seq*2 :] At the reception of a gateway request, the gateway generates a gateway reply and sends it to the gateway requester.

[*Seq*3 :] At the advertisement period, the gateway determines the expected zone of the requesting vehicle using the vehicle's previous location and speed.

[*Seq*4 :] The gateway determines the Proactive Advertisement Zone PAZ.

[*Seq*5 :] The gateway sends the advertisement message. Only vehicles in the PAZ will propagate the advertisement message.

Figure 3.3: An illustrative example for the gateway discovery and communication protocol

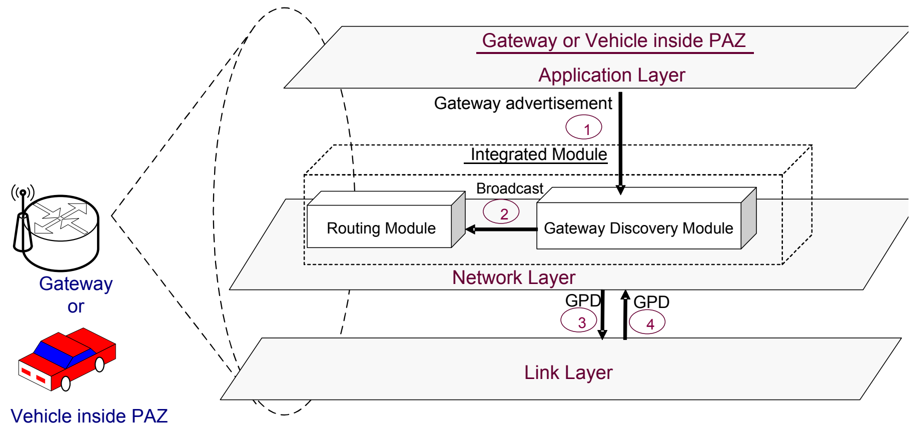
slightly different from its real coordinates; because the location information provided by the GPS is not exact, it has a small margin for error. In our work, we assume that every vehicle knows its exact location. We use this assumption in our analysis but our idea can be applied even if the location information is not exact.

In our protocol, the determination of the Proactive Advertisement Zone (PAZ) of gateways and the determination of PAZ's membership, based on the location information of vehicles, are key components for the design of scalable gateway discovery protocols. Our LAGAD algorithm consists of a lightweight advertisement and a discovery technique that aims to save network resources by finding a gateway as well as its routing information. In large-scale VANets, redundant flooding due to gateway advertisement and discovery can engender network congestion and can affect drastically the limited resources in vehicular networks. If we integrate the gateway discovery information in the routing discovery information, then an efficient network layer based service discovery protocol can be obtained.

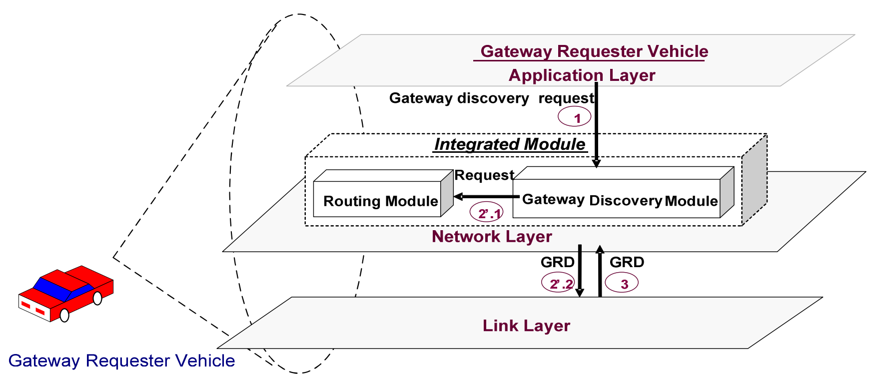
As routing protocol, we use the Connectionless Approach for streets (CLA-S) [40] which is a packet based routing protocol and we use UDP for data communication. The routing protocol CLA-S [40] is robust and solves the link break issue. In fact, in CLA-S routes are not established in a hop by hop basis and communication packets are retransmitted by all nodes inside a predetermined forwarding area. Thus, multiple geographic paths are provided to relay data from the source to the destination.

In our protocol, we assume that the VANet is secure. However, we believe that some security issues like blackhole attacks are very important and we decided to handle this issue in our protocol. For this purpose, we modified the CLA-S [40] routing protocol slightly in order to prevent blackhole attacks. Thus, we assumed that every node in the forwarding zone retransmits the forwarded packet, as opposed to the original CLA-S where a node decides to retransmit or not only after a certain delay. Consequently, blackhole intermediate vehicles that could redirect the traffic could be prevented.

Our LAGAD protocol is based on a hybrid gateway discovery model that permits each



(a) Gateway advertisement process



(b) Gateway discovery process

Figure 3.4: Gateway advertisement and discovery processes

vehicle to perform proactive gateway discovery if it is inside the location-aided advertisement zone of a gateway, and a reactive discovery if it is outside that zone.

Figure 3.4 provides a simplified architecture of the framework resulting from the integration of the gateway discovery module (see Figure 3.4(a)) and the routing module (see Figure 3.4(b)) in the network layer. It provides an overview of the collaboration between the routing module and the gateway discovery module for gateway advertisement and discovery.

There are two main parts in our protocol. The first part is executed at the gateway as illustrated in Algorithm 3.2.1. The second part is executed at the requesting vehicle or the

Algorithm 3.2.1 PROTOCOL AT THE GATEWAY GI

Case A. Initially:
Step A0. G_i broadcasts advertisement messages *GAdv_msg* periodically in one hop.
Case B. G_i receives a gateway request message *GReq_msg* from V_j:
Step B0. G_i sends a gateway reply message *GRep_msg* to the source vehicle V_j (CLA-S)
Case C. G_i receives a $Flow - ID$ message containing the location update information from V_j:
Step C0. Send $Flow - ID$ messages through the Internet
Step C1. Wait a period of time
Step C2. G_i determines the expected zone of V_j.
while Step C3. the expected zone of V_j is relevant **do**
    Step C3.0 G_i determines the request zone of V_j.
    Step C3.1 G_i sends advertisement messages *GAdv_msg* in the request zone of V_j (Geocast)
    Step C3.2 Wait a period of time.
    Step C3.3 G_i determines the expected zone of V_j.
end while
Case D. G_i receives a $Flow - ID$ messages to be sent to V_j:
Step D0. G_i determines the expected zone of V_j.
Step D1. G_i determines the request zone of V_j.
Step D2. G_i sends $Flow - ID$ messages in the request zone of V_j. (Geocast)

intermediate vehicles as shown in Algorithm 3.2.2, and Algorithm 3.2.3, respectively. In the following, we describe our protocol at the gateway and at the vehicle.

- **Protocol at the Gateway:** Our algorithm at the gateway G_i is summarized in Algorithm 3.2.1. Figure 3.4(a) illustrates the gateway advertisement process. Initially, gateways advertise themselves periodically, by sending Gateway Advertisement (GAdv) messages in one hop (Case A). When the required gateway receives the request message, it generates a Gateway Reply (GRep) message and sends it to the original source vehicle (Case B) using CLA-S [40]. The gateway's reply message contains the subnet prefix of the gateway that will be used later by the source vehicle to send data through the gateway. It also contains the location of the gateway. Then, the gateway sends Gateway Advertisement (GAdv) messages in its advertisement zone. In our protocol, we determine two types of zones: the *expected zone* and the *gateway advertisement zone*. The expected zone comprises the road sections where the source vehicle could be located when the gateway sends the GAdv message. The gateway can determine the expected zone by using the location information and the average speed of the source vehicle. In order to determine the expected zone of a source vehicle at the current time t_1, we assume that the gateway recognizes the location $L_0(x_0, y_0)$ of the source vehicle S at time t_0. This information is collected from the Gateway Request (GReq) message

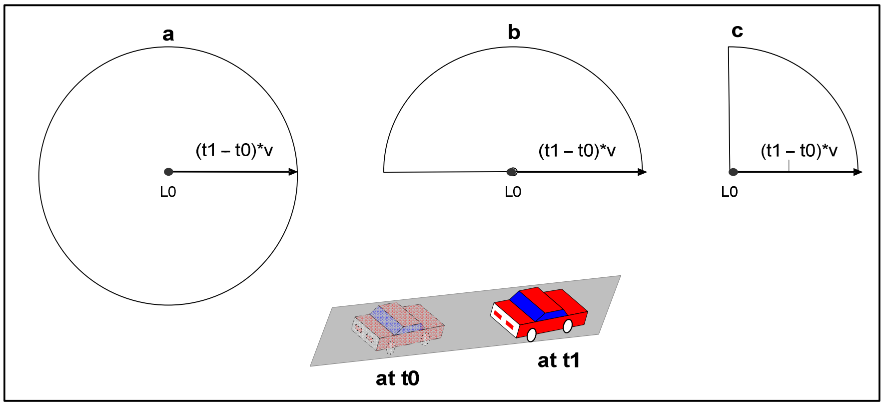
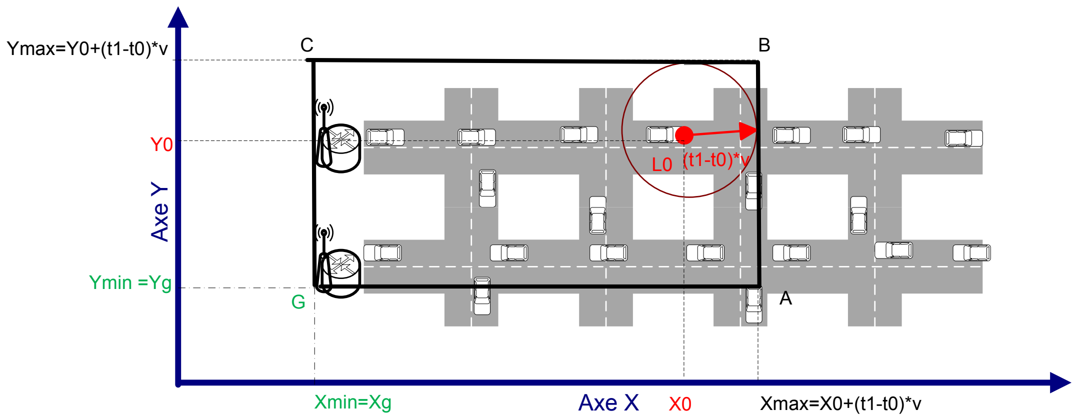


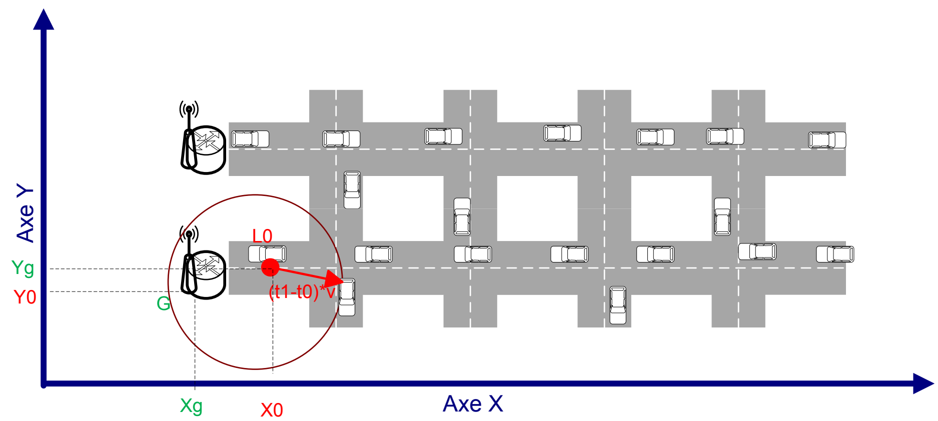
Figure 3.5: Possible expected zones

sent by the source vehicle. The expected zone of the source vehicle S at time t_1 from the point of view of the gateway G is the area expected by G to contain the source vehicle S at time t_1. To determine the expected zone of the source vehicle S at time t_1, the gateway uses its knowledge about location L_0 of the source vehicle S at time t_0. In addition, if G knows the average speed v of the vehicle S, then it can determine the expected zone of S at t_1 as the road sections included in the circle of radius $(t_1 - t_0)v$ and centered at the location L_0 (see Figure 3.5). We have assumed the average speed v of S. However, if the speed of S is larger than the average speed v, the source vehicle will be outside the expected zone. Consequently, the expected zone is only an estimation by the gateway G of the possible current location of the source vehicle S. Yet, it is possible to use the maximum speed or other speed distributions instead of the average speed.

The gateway advertisement zone is the minimal zone containing the gateway and the expected zone. It comprises all the road sections between the gateway and the expected location of the source vehicle. The gateway advertisement zone has a rectangular shape delimited by the lines $D_{Xmin}(x_g = Xmin)$, $D_{Xmax}(x_0 = Xmax)$, $D_{Ymin}(y_g = Ymin)$ and $D_{Ymax}(y_0 = Ymax)$ where $Xmin = minimum(xg, x_0 - (t_1 - t_0)v)$, $Xmax = maximum(xg, x_0 + (t_1 - t_0)v)$, $Ymin = minimum(yg, y_0 - (t_1 - t_0)v)$, and $Ymax = maximum(yg, y_0 + (t_1 - t_0)v)$. Importantly, this scheme is applied only when the gateway



(a) Vehicle outside PAZ



(b) Vehicle inside PAZ

Figure 3.6: The Proactive Advertisement Zone PAZ

G is outside the expected zone of the source vehicle S. In fact, if the gateway G is inside the expected zone of S, then the proactive advertisement zone and the expected zone are the same. In Figure 3.6(a), the proactive advertisement zone is the rectangle delimited by D_{Xmin}, D_{Xmax}, D_{Ymin} and D_{Ymax}. However, in Figure 3.6(b), the proactive advertisement zone is only the circle C. When the gateway G sends its advertisement message (GAdv), it includes $Xmin$, $Xmax$, $Ymin$, and $Ymax$ so that every other vehicle can determine whether or not it is in the PAZ of G.

The gateway advertisement message is sent periodically in the advertisement zone after the expected zone of the source vehicle is determined (Case C). The gateway efficiently determines the gateway advertisement zone; in some cases, when source vehicles are close together, the expected zones can overlap. In this case, the gateway merges with the overlapping expected zones for a good computation of an expected zone that covers more than one source vehicle. In our protocol, the advertisement zone of the gateway is adjusted periodically. It depends on the initial location of the source vehicle and its average speed. The gateway advertisement zone is determined every time the gateway has to send an advertisement message. It is worthy to note that the PAZ is proportional to the average speed v of the source vehicle and to the time elapsed since the last update of the location information of the source vehicle S. A large proactive advertisement zone would increase the control overhead. This could be used to assess the tradeoff between the advertisement frequency and the size of PAZ.

When the gateway receives a Gateway Reactive Discovery (GRD) packet, the integration module decodes the packet. This permits the separation of routing information from gateway information that will be treated differently by the routing module and the gateway discovery module respectively.

- **Protocol at the requesting vehicle:** Our algorithm at the requesting vehicle is summarized in Algorithm 3.2.2. Figure 3.4(b) illustrates the process of a gateway discovery protocol. When an application program installed at a vehicle wants to find a gateway, it checks whether a gateway exists or not in its gateway information table. If a gateway does not exist in its gateway information table, the gateway discovery module then requests the routing module to collaborate in order to find a gateway and sends the application request to the integrated module. The routing module generates the appropriate routing request packet, adds the current identifier of the vehicle and sends the routing request packet to the Integrated Module. This latter generates the appropriate GRD packet or the Gateway Request (GReq) message. The GRD packet contains both discovery and routing information, it contains the location and the average speed of the source vehicle as well. The GRD packet is propagated in the VANet. When the source vehicle S sends

Algorithm 3.2.2 PROTOCOL AT THE REQUESTING VEHICLE VJ

```
Case E. Vj needs to find a gateway Gi
while Step E0. (N<3 or NO GRep_msg is received) do
    Vj broadcasts a request message GReq_msg in a predetermined search area comprising x hops.
end while
if Step E1. (GRep_msg received) then
    Step E1.0. GOTO Step F0.
else
    Step E1.2. Exit: Gateway not found
end if
Case F. Vj receives a reply message GRep_msg from Gi
Step F0. Store gateway information
Step F1. Wait the estimated time by Vj to receive all replies from all the gateways in the search area of Vj.
if (Vj needs to send a flow through a gateway Gi) then
    GOTO Step G0.
end if
Case G. Vj needs to send a flow through a gateway Gi
Step G0. Choose the appropriate gateway Gi.
if Step G1. (an appropriate gateway is found) then
    Send Flow − ID and the location update information in a message to the chosen Gi (CLA-S)
else
    GOTO Step E0.
end if
Case H. Vj receives a location-aided advertisement message from Gi
Step H0. Store gateway information of Gi.
Case I. Vj receives a Flow_ID message from Gi
Step I0. Store the flow
```

its data through the gateway G, it includes its most recent location information and average speed v so that the gateway G can update the information of the source vehicle S for future advertisements.

- **Protocol at an Intermediate Vehicle:** Our algorithm at an intermediate vehicle is summarized in Algorithm 3.2.3. In our Location-Aided Gateway Advertisement algorithm,

Algorithm 3.2.3 PROTOCOL AT THE INTERMEDIATE VEHICLE VK

```
Case J. Vk receives a reply message GRep_msg from Gi to Vj
Step J0. Forward the message to its destination Vj (CLA-S)
Case K. Vk receives a location-aided advertisement message from Gi
Step K0. Store gateway information.
if Step K1. (Vk is in PAZ and message is not redundant) then
    Forward Gateway information
end if
Case L. Vk receives a Flow_ID msg from Gi to Vj
if Step L0. (Vk is in the PAZ of Gi and msg is not redundant) then
    Forward the flow to Vj
end if
Case M. Vk receives a Flow_ID message from Vj to Gi
Step M0. Forward the flow to Gi (CLA-S)
Case N. Vk receives a location-aided request message
Step N0. Forward Gateway request message.
```

a vehicle in the PAZ either forwards or discards any Gateway Advertisement (GAdv) message it receives. Consequently, in our implementation of LAGAD, a vehicle must be able to determine whether or not it is in the proactive advertisement zone for a particular gateway.

When an intermediate vehicle I_1 in the specified rectangular PAZ receives the GAdv message, it forwards it. However, when the intermediate vehicle I_2 receives the GAdv message and is not in the specified rectangular PAZ, it discards the message.

The integrated module of a receiving vehicle eliminates redundant packets, decodes the GPD packet and separates the gateway advertisement information from the routing information. Upon the reception of a GRD packet by a vehicle, the packet is decoded by the integration module. The Gateway Module checks whether valid gateway information exists in its Gateway Table. The GRD packet continues to be propagated in the VANet until a vehicle finds the gateway information in its gateway table or until the GRD reaches a gateway. In both cases, a gateway reply message is generated and is sent back to the gateway requester using the CLA-S [40] routing protocol. Intermediate vehicles forwarding the gateway reply message, that can be used in future discovery queries, cache the gateway information in their gateway tables.

3.2.4 Multi-Channels and Multi-Interfaces LAGAD

In what follow, we shall explain how our LAGAD protocol benefits from the channel diversity feature. Vehicular network capacities and capabilities are improved when an efficient utilization of channels is taken into consideration. In our LAGAD protocol, we implement the multi-channels and multi-interfaces feature very simply and easily. We assume that every vehicle is equipped with two interfaces (*int*1 and *int*2). Every interface uses a fixed channel different from the one used in the other interface of the same vehicle. In order to guarantee the connectivity in the vehicular network and to avoid channel switching costs, we assume that *int*1 is set to the fixed channel1 and that *int*2 is set to the fixed channel2. The usage of the different interfaces is handled at the routing module. The latter determines which interface is used for the propagation of LAGAD messages as shown in Figure 3.2(b). We manage the usage of interfaces in such a way that sending and receiving messages are performed using different channels.

3.3 Proof of Correctness

The proof of correctness of the gateway discovery protocol can be done in three steps. First, we need to prove that the discovery protocol is accurate by showing that the returned result matches effectively the user's request. Second, we have to show that the discovery protocol is complete by showing that the protocol finds all the services that satisfy the query. Third, we need to prove that our discovery protocol terminates in a finite time and that there are no deadlocks.

Lemma 3.3.1. ***Relevance of the expected zone****: The expected zone of V computed by a gateway G at time t_1 is said to be relevant if and only if the following equation holds:*

$$\left(\sqrt{(x_0 - x_g)^2 + (y_0 - y_g)^2}\right) + ((t_1 - t_0)v) \leq max\_cov \tag{3.2}$$

where V is a vehicle located at location $L_0(x_0, y_0)$ at t_0 moving with an average speed v and G is a gateway with a fixed location $L_g(x_g, y_g)$ and a maximum coverage max_cov.

Proof: We define the variable *max_cov* as the maximum distance determined by a gateway to serve vehicles in multihops. It is important to notice that *max_cov* does not stand for the one hop radius coverage of the gateway. The distance *max_cov* permits to the gateway to limit the number of multihops in order to avoid overloading the bandwidth. We assume that only a percentage α of the total bandwidth is used for gateway advertisement. By knowing the size of gateway advertisement message *GAdv_msg_size*, we can determine the maximum number of advertisement messages *maxn_GAdv_Cg* allowed to be exchanged between vehicles in the

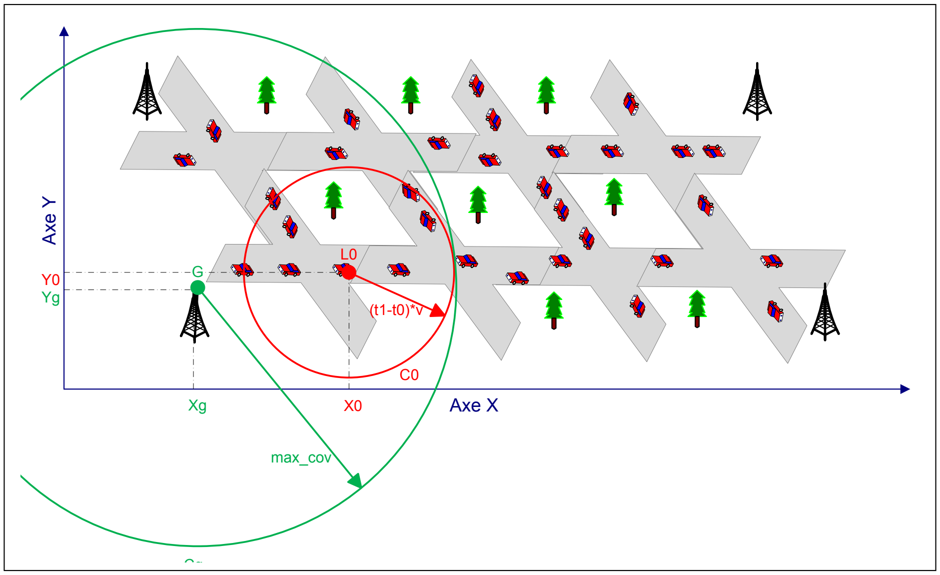


Figure 3.7: Determination of the relevance of the expected zone

gateway area delimited by Cg in Figure 3.7, such that the bandwidth requirements are met.

$$maxn\_GAdv\_Cg = \frac{\alpha * Bandwidth}{GAdv\_msg\_size} \tag{3.3}$$

In the Proactive Advertisement Zone (*PAZ*) of a gateway in our protocol, the propagation of the advertisement messages follows a controlled flooding, which eliminates the propagation of redundant messages. Hence, the same advertisement message is propagated only once by a vehicle, even if it is received several times as in *Step K1*. Consequently, the total number of gateway advertisement messages during one period in *PAZ*, delimited by C_g, is equal to the number of vehicles in the *PAZ*.

$$maxn\_GAdv\_Cg = nVehicles\_Cg \tag{3.4}$$

Assuming the density ρ (number of vehicles per meter square), we can determine the number of vehicles in C_g after computing the area of C_g and the number of road sections in C_g. In order to determine the number of road sections in C_g, we use the cross-multiplication. Assuming the number of road sections nRS in the whole vehicular network area of surface

$Snet$, the number of road sections nRS_Cg in the circle C_g is:

$$nRS\_Cg = \frac{nRS \times \Pi \times max\_cov^2}{Snet} \tag{3.5}$$

Thus, the number of vehicles in C_g is:

$$nVehicles\_Cg = nRS\_Cg \times sRS \times \rho \tag{3.6}$$

where sRS is the surface of a road section. Thus,

$$nVehicles\_Cg = \frac{nRS \times \Pi \times max\_cov^2}{Snet} \times sRS \times \rho \tag{3.7}$$

We deduct from Equations 3.3, 3.4 and 3.7 that

$$\frac{nRS \times \Pi \times max\_cov^2}{Snet} \times sRS \times \rho = \frac{\alpha * Bandwidth}{GAdv\_msg\_size} \tag{3.8}$$

Consequently:

$$max\_cov = (\frac{\alpha * Bandwidth \times Snet}{GAdv\_msg\_size \times \Pi \times \rho \times nRS \times sRS})^{\frac{1}{2}} \tag{3.9}$$

After a gateway determines its max_cov, it can check whether or not the expected zone of a requesting vehicle is relevant or not. For this purpose, the distance between the gateway and the initial location of the requesting vehicle at time $t0$, plus the distance between the initial location of the requesting vehicle and its expected location at time $t1$, when the gateway decides to send an advertisement message, should not exceed max_cov, knowing that the vehicle is moving with the average speed v. We can first determine the distance between the gateway and the initial location of the requesting vehicle d_{LgL0}, knowing the location of the gateway $Lg(x_g, y_g)$ which is fixed and the location of the vehicle at t_0 $L_0(x_0, y_0)$ and its average speed v:

$$d_{LgL0} = (\sqrt{(x_0 - x_g)^2 + (y_0 - y_g)^2}) \tag{3.10}$$

Secondly, the distance covered by the vehicle at the time t_1 which is the radius of $C0$, R_{C0}:

$$R_{C0} = (t_1 - t_0)v \tag{3.11}$$

In order to be sure that a requesting vehicle is still inside the gateway multihop coverage, $d_{LgL0} + R_{C0}$ must be less or equal to max_cov, hence we derive our Lemma 3.3.1.

Lemma 3.3.2. ***(Gateway Discovery Accuracy):*** *The Location-Aided Gateway Advertisement and Discovery protocol finds a gateway to the requesting vehicle in a finite time, if there is a gateway in the search area of the requesting vehicle.*

Proof: Let us assume that the Location-Aided Gateway Advertisement and Discovery protocol cannot find a gateway to the requesting vehicle in a finite time even though there is a gateway G_i in the search area of V_j. The requesting vehicle keeps sending requests for three rounds in its search area. If a gateway receives a request it sends a reply as in *Step B0.* Thus, the requesting vehicle will receive a gateway reply. Therefore, V_j will eventually find a gateway in a finite time. This is a contradiction. Therefore, our assumption is false and the lemma is proved.

Lemma 3.3.3. ***(Gateway Discovery Completeness):*** *Assume that there are many gateways in the vicinity of a requester, the requester will receive replies from all the gateways in its vicinity in a finite time.*

Proof: Let us assume that the Location-Aided Gateway Advertisement and Discovery protocol cannot find all the gateways located in the search area of the requesting vehicle V_j in a finite time, even though all the vehicles in the search area of V_j are connected. Since all the vehicles in the search area of V_j are connected, the *GReq_msg* will be received by all the gateways in the predetermined search area. Every gateway that receives the *GReq_msg* will reply with *GRep_msg*, as in *Step B0*. Reply messages will be geocast to the requesting vehicle. Therefore, knowing that vehicles in the search area are connected, the requesting vehicle will eventually receive reply messages from all the gateways in the predetermined search area in a finite time. This is a contradiction. Therefore, our assumption is false and the lemma is proved.

Lemma 3.3.4. ***(Gateway Discovery Correctness):*** *Our Location-Aided Gateway Advertisement and Discovery protocol will terminate with a finite time.*

Proof: The correctness of LAGAD is proved if our proposed protocol terminates the discovery process with a finite time. There are two loops in the Location-Aided Gateway Advertisement and Discovery protocol. The first one is between *Step C3* and *Step C3.4* of the gateway pseudo-code and the second one is between *Step E0* and *Step E1* in the requesting vehicle pseudo-code.

According to Lemma 3.3.1, the expected zone of V_j is relevant if and only if Equation 3.2 holds. In the first loop, *Step C3* tests the expected zone of V_j. Assuming that the corresponding gateway did not receive any update since t_0, the number computed in the left-hand-side of Equation 3.2 increases with the increase of the time t_1. In *Step C3.2*, the gateway waits for a period of time, thus t_1 increases in every round. Consequently, with the increase of the elapsed time, the computed left-hand-side-number in Equation 3.2 will bypass the predetermined *max_cov* of the considered gateway. Hence, the test in *Step C3* will not hold anymore and the first loop terminates in a finite time.

According to Lemma.4.4.1, the second loop of the Location-Aided Gateway Advertisement and Discovery protocol returns a gateway reply message *GRep_msg* in a finite time, if there is at least one gateway in the search area of the requesting vehicle. If there is no gateway in the predetermined search area after three attempts to find a gateway, *Step E1.2* allows the process to be terminated. Hence, either the second loop finds a gateway or terminates in *Step E1* in a finite time.

3.4 Communication and Time Complexities

In this section, we will evaluate the costs of our *LAGAD* protocol in terms of communication and time complexities. Before we proceed further, let us prove in Lemma 3.4.1 how we determine the areas of the proactive advertisement zone *(PAZ)* that will help us in the computation of the complexities.

Lemma 3.4.1. ***(The Proactive Advertisement Zone (PAZ) Area)****: The area of the PAZ of gateway G is:*

$$Area\_PAZ = (X\_max - X\_min) \times (Y\_max - Y\_min) \tag{3.12}$$

where V is a vehicle located at location $l_0(x_0, y_0)$ and at t_0 moving with a speed v, G is a gateway that has a fixed location $l_g(x_g, y_g)$, $t1$ is the starting time of the next advertisement round, $Xmin = minimum(x_g, x_0 - (t_1 - t_0)v)$, $Xmax = maximum(x_g, x_0 + (t_1 - t_0)v)$, $Ymin = minimum(y_g, y_0 - (t_1 - t_0)v)$, and $Ymax = maximum(y_g, y_0 + (t_1 - t_0)v)$.

Proof: Let us consider the area of the Proactive Advertisement Zone *PAZ* illustrated in Figure 3.6(a). The gateway advertisement zone is the minimal zone containing the gateway and the expected zone. It comprises all the road sections between the gateway G with the fixed coordinates (x_g, y_g) and the expected location of the source vehicle at the instant t_1, having the coordinates (x_0, y_0) at the initial time t_0. We represent the expected zone of V by the circle centered at $L0$, which has a radius equal to the value of the maximum distance that could be traversed by the vehicle V at instant t_1 with the average speed v. Thus the circle C is denoted by $C(L0, (t1 - t0)v)$. The gateway advertisement zone has a rectangular shape delimited by the lines $(D_{x=X\_min})$, $(D_{x=X\_max})$, $(D_{y=Y\_min})$ and $(D_{y=Y\_max})$, where $X\_min = minimum(x_g, x_0 - (t_1 - t_0)v)$, $X\_max = maximum(x_g, x_0 + (t_1 - t_0)v)$, $Y\_min = minimum(y_g, y_0 - (t_1 - t_0)v)$, and $Y\_max = maximum(y_g, y_0 + (t_1 - t_0)v)$. Consequently, $Area\_PAZ$ is:

$$Area\_PAZ = (X\_max - X\_min) \times (Y\_max - Y\_min) \tag{3.13}$$

Lemma 3.4.2. *(**Number of gateway advertisement messages**): The number N1 of gateway advertisement messages is computed as follow:*

$$N1 = \frac{nRS \times (X\_max - X\_min) \times (Y\_max - Y\_min)}{Snet} \times sRS \times \rho \times fAdv \times \Delta\_Adv \quad (3.14)$$

where $Lg(x_g, y_g)$ is the location coordinate of the gateway G, $L0(x_0, y_0)$ is the coordinate of the initial location of the requesting vehicle V at t_0, v is its average speed at t_0, t_1 is the time at which the gateway starts the advertisement round, nRS is the number of road segments in the vehicular network, $Snet$ is the surface of the vehicular network, sRS is the surface of a road segment, ρ is the average of a road segment which we assume is also the average density in the $VANet$, $fAdv$ is the frequency of gateway advertisement, Δ_Adv is the whole simulation period, $Xmin = minimum(x_g, x_0 - (t_1 - t_0)v)$, $Xmax = maximum(x_g, x_0 + (t_1 - t_0)v)$, $Ymin = minimum(y_g, y_0 - (t_1 - t_0)v)$, and $Ymax = maximum(y_g, y_0 + (t_1 - t_0)v)$.

Proof: Let us compute the number of advertisement messages initiated by gateways in the VANet. In our gateway discovery protocol, the number of gateway advertisement messages depends mainly from the proactive advertisement zone (*PAZ*). The *PAZ* in our protocol is variable, it depends on the expected zone of the vehicle. Assuming the density ρ (number of vehicles per meter square), we can determine the number of vehicles in the *PAZ* after computing the area of the *PAZ* and the number of road sections in the *PAZ*.

Lemma 3.4.1 provides the area of the PAZ. To determine the number of road sections in the PAZ, we used cross-multiplication. Assuming the number of road sections nRS in the whole vehicular network area of surface $Snet$, the number of road sections nRS_PAZ in the proactive advertisement zone area PAZ_Area is:

$$nRS\_PAZ = \frac{nRS \times PAZ\_Area}{Snet}. \quad (3.15)$$

The number of vehicles in the *PAZ* is:

$$number\_of\_vehicles\_in\_PAZ = nRS\_PAZ \times sRS \times \rho \tag{3.16}$$

where sRS is the surface of a road section and it is a predetermined fixed value.

The dissemination of the gateway advertisement messages in the *PAZ* is controlled in order to avoid redundancy. Thus every vehicle in the *PAZ* retransmits the advertisement messages only received for the first time; it ignores redundant advertisement messages that it has transmitted already. Following this controlled dissemination, we can conclude that every vehicle in the *PAZ* will transmit the same advertisement message once. Hence:

$number\_of\_advertisement\_messages\_in\_one\_period = number\_of\_vehicles\_in\_PAZ \Leftrightarrow$

$number\_of\_advertisement\_messages\_in\_one\_period = nRS\_PAZ \times sRS \times \rho.$

Since the gateway advertisement messages are propagated periodically, during the period Δ_Adv and assuming that the frequency of advertisement during the simulation period is $fAdv$, we can incur that:

$$N1 = nRS\_PAZ \times sRS \times \rho \times fAdv \times \Delta\_Adv \Leftrightarrow$$

$$N1 = \frac{nRS \times PAZ\_Area}{Snet} \times sRS \times \rho \times fAdv \times \Delta\_Adv \Leftrightarrow$$

$$N1 = \frac{nRS \times (X\_max - X\_min) \times (Y\_max - Y\_min)}{Snet} \times sRS \times \rho \times fAdv \times \Delta\_Adv$$

Lemma 3.4.3. *(**Number of request messages**): the number of request messages $N2$ sent by vehicles during the simulation period is:*

$N2 = nRS \times sRS \times \rho + \sum_{i=2}^{nReq}(nRS_i \times sRS \times \rho - ((\frac{nRS_i \times ((X\_max - X\_min) \times (X\_max - X\_min))}{Snet}) \times sRS \times \rho)).$

where $L(xv, yv)$ is the location of the vehicle V, $G(xg, yg)$ is the location of the gateway G and ρ is the density in VANet.

Proof: Let us compute the number of request messages initiated by gateway requesters in the VANet. In the worst case, gateways have not started advertising themselves. Thus the request message is propagated by all the vehicles in the considered vehicular network until it reaches a gateway. This case occurs only for the first gateway request. After that, gateways start advertising themselves and the request messages are intercepted and processed by the vehicles in the border of the PAZ. This is because the dissemination of request messages is controlled in order to avoid sending redundant messages; i.e, a vehicle forwards the same message only once. Hence, the number of request messages will be equal to the number of vehicles in the segments' area where the request message is propagated. Thus, the number of request messages is equal to the number of vehicles in the VANet for the first request, plus the number of vehicles in the VANet except those in PAZs for the other requests:

$N2 = N\_Vehicles\_Vanet + \sum_{i=2}^{nReq}(N\_Vehicles\_Vanet - N\_Vehicles\_PAZs) \Leftrightarrow$

$N2 = nRS \times sRS \times \rho + \sum_{i=2}^{nReq}(nRS \times sRS \times \rho - (nRS\_PAZ \times sRS \times \rho)) \Leftrightarrow$

$N2 = nRS \times sRS \times \rho + \sum_{i=2}^{nReq}(nRS \times sRS \times \rho - (\frac{nRS \times PAZ\_Area}{Snet} \times sRS \times \rho)) \Leftrightarrow$

$N2 = nRS \times sRS \times \rho + \sum_{i=2}^{nReq}(nRS \times sRS \times \rho - ((\frac{nRS \times ((X\_max - X\_min) \times (X\_max - X\_min))}{Snet}) \times sRS \times \rho))$

Lemma 3.4.4. *(**Number of reply messages**): In the worst case, the number of reply messages* $N3$ *for one gateway request is:*

$$N3 = (\frac{(X\_max - X\_min) + (Y\_max - Y\_min)}{lseg}) \times sRS \times \rho \tag{3.17}$$

where $lseg$ is the length of a road segment and it is a predetermined fixed value.

Proof: Let us compute the number of reply messages sent by gateways towards the requesting vehicles. The number of reply messages is mainly the number of vehicles inside the forwarding zone determined through the CLA-S[40]. It is the number of vehicles in the road segments between the gateway or the replying vehicle and the requesting vehicle. In the worst case, the request message reaches the gateway and the latter has to send the reply message through the CLA-S[40]. The number of segments between the gateway and the requesting vehicles is:

$$nSeg = \frac{(X\_max - X\_min) + (Y\_max - Y\_min)}{lseg} \tag{3.18}$$

where *lseg* is the length of a road segment and it is a predetermined fixed value. Consequently, the number of reply messages is:

$$N3 = nSeg \times sRS \times \rho \Leftrightarrow \tag{3.19}$$

$$N3 = (\frac{(X\_max - X\_min) + (Y\_max - Y\_min)}{lseg}) \times sRS \times \rho \tag{3.20}$$

Lemma 3.4.5. *(**Message complexity**): The total number of messages exchanged in our LAGAD protocol is:*

$Message\_complexity = (\frac{nRS \times (X\_max - X\_min) \times (Y\_max - Y\_min)}{Snet} \times sRS \times \rho \times fAdv \times \Delta\_Adv) + (nRS \times sRS \times \rho + \sum_{i=2}^{nReq}(nRS_i \times sRS \times \rho - ((\frac{nRS_i \times ((X\_max - X\_min) \times (X\_max - X\_min))}{Snet}) \times sRS \times \rho))) + \sum_{j=1}^{nReq}((\frac{(X\_max - X\_min) + (Y\_max - Y\_min)}{lseg}) \times sRS \times \rho)$

Proof: We consider that the message complexity depends from the number of gateways and also from the number of requesters. In our protocol, the proactive advertisement zone of the gateway is not fixed, but it is dynamic and adaptive. The message complexity depends on the location of the requesting vehicles and their movements. The total message complexity would be the summation of the number of advertisement messages, the number of request messages and the number of reply messages relevant to each requesting vehicle during the simulation period. The proof is deduced from lemmas 3.4.2, 3.4.3 and 3.4.4.

$Message\_complexity = N1 + \sum_{i=1}^{i=number\_requesters}(N2_i + N3_i) \Leftrightarrow$

$Message\_complexity = (\frac{nRS \times (X\_max - X\_min) \times (Y\_max - Y\_min)}{Snet} \times sRS \times \rho \times fAdv \times \Delta\_Adv) + (nRS \times sRS \times \rho + \sum_{i=2}^{nReq}(nRS_i \times sRS \times \rho - ((\frac{nRS_i \times ((X\_max - X\_min) \times (X\_max - X\_min))}{Snet}) \times sRS \times \rho))) + \sum_{j=1}^{nReq}((\frac{(X\_max - X\_min) + (Y\_max - Y\_min)}{lseg}) \times sRS \times \rho)$

Lemma 3.4.6. *(**Time complexity**): The average time required for a gateway requester to receive the first gateway reply is:*

$TC = ((nRS \times sRS \times \rho + \sum_{i=2}^{nReq}(nRS_i \times sRS \times \rho - ((\frac{nRS_i \times ((X\_max - X\_min) \times (X\_max - X\_min))}{Snet}) \times sRS \times \rho))) + \sum_{j=1}^{nReq}((\frac{(X\_max - X\_min) + (Y\_max - Y\_min)}{lseg}) \times sRS \times \rho))/nReq \times \lambda t$

where λt is the time needed for the propagation of a message in the vehicular network.

Proof: We define the time complexity as the average time required for a service requester to receive the first service reply during the simulation time, where *nReq* requesters initiate the gateway discovery process. In the worst case, the gateway request message generated by the requester has to go until it reaches the gateway. Then the gateway has to generate the gateway reply message and to send it back to the requester. Thus, assuming that: *(i)* $N\_of\_Request\_messages$ is the number of request messages propagated from the requester to the gateway; *(ii)* $N\_of\_reply\_messages$ is the number of reply messages propagated from the gateway to the requester; and (iii) λt is the propagation time of a message in the VANet, then the time required until a requester gets its first gateway reply message would be:
$N\_of\_Request\_messages + N\_of\_reply\_messages \times \lambda t$. Since we have *nReq* requesters that have initiated the gateway discovery mechanism, then the average time required for a gateway requester to receive the first gateway reply is:

$$TC = (\sum_{i=1}^{i=nReq}(N\_of\_Request\_messages\_i + N\_of\_reply\_messages_i))/nReq \times \lambda t \Leftrightarrow$$

$$TC = ((nRS \times sRS \times \rho + \sum_{i=2}^{nReq}(nRS_i \times sRS \times \rho - ((\frac{nRS_i \times ((X\_max - X\_min) \times (X\_max - X\_min))}{Snet}) \times sRS \times \rho))) + \sum_{j=1}^{nReq}((\frac{(X\_max - X\_min) + (Y\_max - Y\_min)}{lseg}) \times sRS \times \rho))/nReq \times \lambda t$$

3.5 Different Discovery Approaches and Mobility Models: A Comparison

In this section, we evaluate the performance of our protocol and compare our scheme with different existing gateway discovery approaches: Reactive [77], Proactive [77], and Hybrid [70] approaches. We performed our experiments in a large-scale vehicular network and we used a Manhattan road-traffic pattern.

3.5.1 Simulation Setup

In order to be fair in our comparison, we modified the existing proactive, reactive and hybrid existing discovery approaches [77, 70], such as they are network layer-based and use the channel diversity mechanism. We choose to evaluate the LAGAD's performance using mobility traces that correspond to a *Manhattan* traffic pattern. In our simulation experiments, we compared our adaptive hybrid approach to a reactive approach [77], a proactive approach [77], and an hybrid approach [70] with different fixed radius (1, 2, and 3 hops). We used a large-scale vehicular network.

We choose to draw a comparison among these different approaches in a large-scale vehicular network because we want to prove that (i) our approach outperforms the reactive and the proactive approaches in some performance metrics; (ii) our adaptive hybrid location-aided gateway discovery approach outperforms standard hybrid gateway discovery approaches in some performance metrics; and (iii) our approach scales well with the increase in the number of gateway requests.

We use the simulation environment illustrated in Table 4.3. The density of the network is 125 vehicles per kilometer square, the simulation time is 1200 seconds, the number of gateway clients is 100 and the number of requests range between 100 and 1000 requests. In the course of our experiments, we define the request number as the number of gateway queries sent by gateway clients. A gateway client can generate many gateway requests. We choose to evaluate the performance of our algorithm using the following metrics:

- *Success rate*: indicates the average fraction of successful LAGAD transactions;
- *Response time*: determines the average time of successful LAGAD transactions. It measures the elapsed time for getting a valid gateway reply in response to a gateway request sent by a vehicle. This metric takes into account several factors such as transmission and message processing delay, just to mention a few;
- *Bandwidth usage*: measures the overhead of our algorithm in terms of the number of messages needed to satisfy the driver's gateway requests;
- *Average bandwidth usage per request*: calculates the average bandwidth used per one successful gateway request.

Table 3.1: Simulation environment for the city scenario

Parameter name	Parameter value
Wireless medium	802.11
Data transmission rate	11Mbps
Transmission range (meters)	100
Received signal strength threshold (meters)	100
Vehicle's Speed (meter/second)	[0..20]
Gateway advertisement's life time (seconds)	30
Simulation time(seconds)	1200
Simulation area ($meter^2$)	1 440 000
Density (vehicles/$kilometer^2$)	120
Gateway number	9
Advertisement frequency (message/second)	1/50
LAGA's speed	20
Maximum gateway coverage (meters)	250
Gateway clients' number	100

- *Data packets delivery rate*: computes the fraction of data packets delivered to the destination and the data packets sent by the sources;
- *End-to-end delay*: indicates in seconds the time it takes a data packet to reach the destination gateway from the source or vice versa. It includes the processing time, queuing delay, retransmission delay at the MAC and the propagation and retransmission times;
- *Packet order*: indicates whether or not the data packets are received in the same order as their transmission;
- *Duplicate data packets*: determines the average number of duplicated data packets received by the destination.

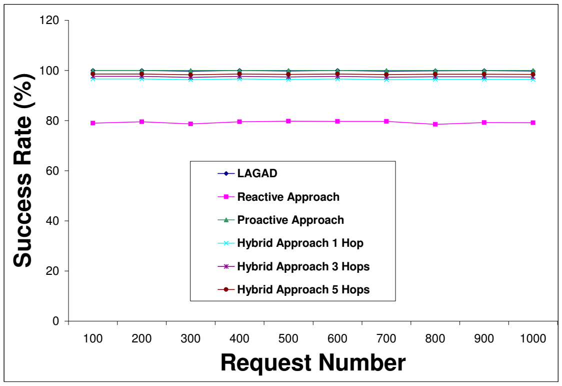
3.5.2 Simulation Experiments

In what follows, we report the simulation experiments we compared to our algorithm, using the reactive approach, the proactive approach and the hybrid approach in a large-scale setting. We conducted an extensive set of experiments in the network simulator NS2 [4]. During the course of our experiments, we varied the number of service requests between 100 and 1000 requests. Service requests' initial time was generated randomly. The following experimental data were obtained by averaging several runs with a confidence of 95 percent. We used a different seed for every run. The results are summarized in graphs in Figure 3.8.

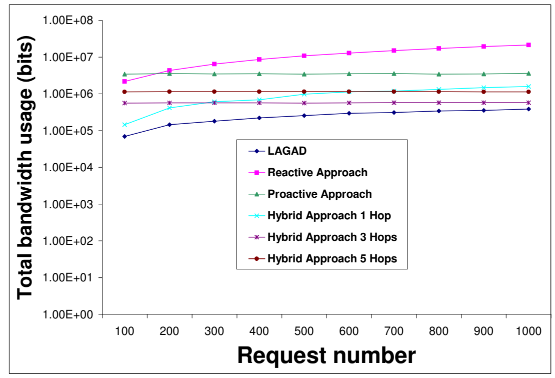
The success rate for the large-scale VANet is shown in Figure 3.8(a). As we can see, our scheme, the proactive scheme and the hybrid schemes have almost the same success rate for a number of gateway requests between 100 and 1000, and their success value is quite high (more than 96 percent). In addition, our LAGAD protocol provides high success rates, thanks to its advertisement zone adaptation mechanism. LAGAD has no obligation to flood the whole network with advertisement messages like the proactive approach and does not have to use a fixed, large advertisement zone like the hybrid approaches. The reactive discovery scheme has the lowest success rate. This is due to the high mobility of vehicles in VANets. In such high mobile environments, network partitions could be incurred. Thus, the gateway request could not reach the gateway (as it is the case for the reactive approach). In the proactive and hybrid approaches, gateways are passively registered in the tables of the gateway information belonging to potential gateway clients. Thus, a gateway discovery query has a likelihood of being higher answered by the requesting client itself or intermediate vehicles. However, there is no guarantee that the routing path to the gateway is up to date, due to the high mobility of vehicles in the VANet as depicted in Figure 3.8(f).

Figure 3.8(d) portrays the results obtained for the average response time per driver's requests using the Manhattan traffic model in a large-scale setting. We noticed that proactive and hybrid schemes achieve a low response time (in the order of milliseconds). This is mainly due to the multi-channel, multi-interface usage which reduces the congestion on single channels and reduces the transmission time of messages. Figure 3.8(e) shows how the channel diversity usage could improve the response time for our LAGAD.

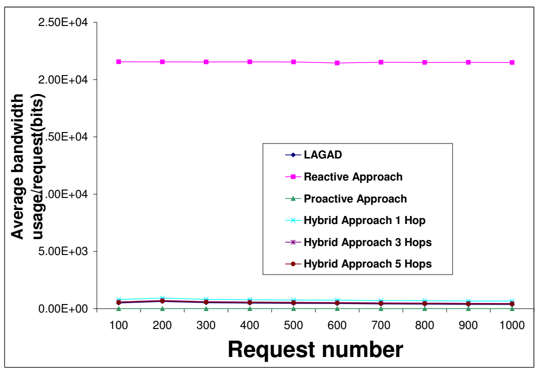
The reactive approach in a large-scale setting has the highest response time among others. The response time of the reactive approach is the highest among other approaches because a gateway request has to traverse hop by hop many vehicles to reach the gateway and then a reply has to be sent back from the gateway. If the gateway is far away from the requester, the



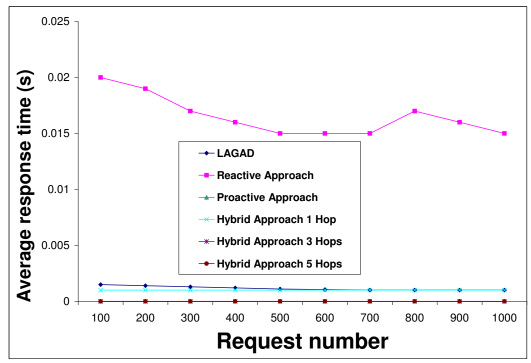
(a) Success Rate comparison



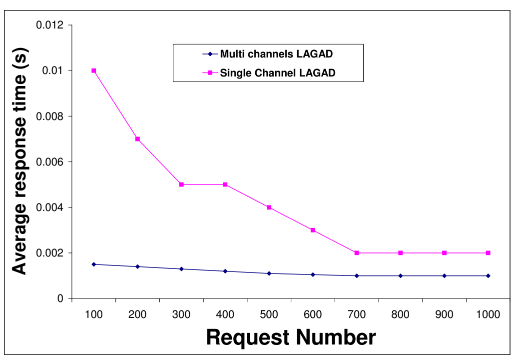
(b) Bandwidth usage comparison



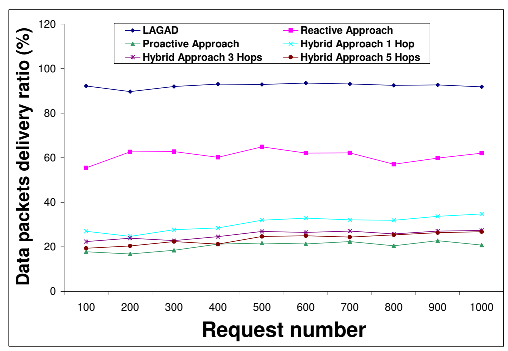
(c) Average per request Bandwidth usage comparison



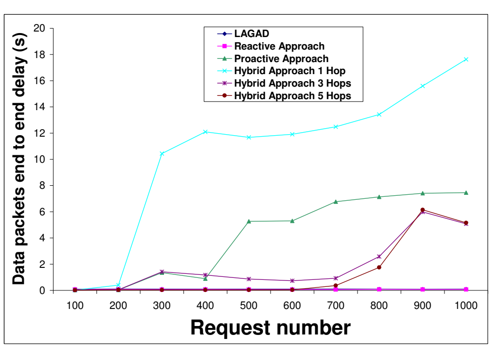
(d) Average response time comparison



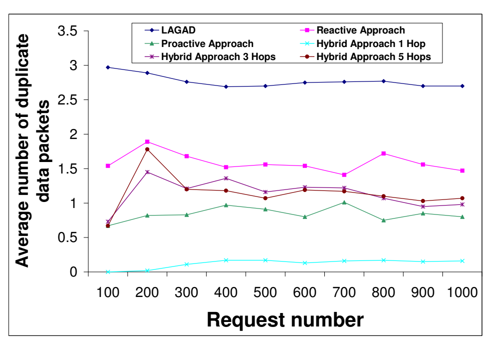
(e) Response time of multi channels LAGAD to the single channel LAGAD



(f) Data packets delivery rate comparison



(g) Data packets end to end delay comparison



(h) Average data packet duplicates' comparison

Figure 3.8: Performance evaluation in large-scale Manhattan traffic scenario

transmission time of the request and the reply messages are high, which can increase the response time of the reactive approach. The proactive approach and the hybrid approaches with five and three hops respectively have the lowest response time. We noticed from Figure 3.8(d) that the response time for these approaches is almost negligible for a number of requests ranging from 100 to 1000. This is mainly because for proactive and hybrid approaches with a large advertisement zone, gateway advertisement messages are flooded in the whole network or in a large advertisement zone. Thus, vehicles know about existing gateways passively, but at the expense of a high bandwidth usage as depicted in Figure 3.8(b).

The hybrid approach with 1 hop has a higher response time than the hybrid approach with five and three hops respectively, because with a restricted advertisement zone, the transmission time of gateway requests and replies takes more time than in the case of a large advertisement zone. Our LAGAD has a response time in the order of 1 ms for a number of requests ranging between 100 and 1000 requests. Our LAGAD succeeds to adjust its location-aided advertisement zone and the average response time is maintained to its best value equal to 1ms in our simulations. It achieves a low response time while guaranteeing a low bandwidth usage as opposed to the proactive and the hybrid approach with a large advertisement zone. This proves the scalability of LAGAD.

Figure 3.8(b) portrays the bandwidth usage of the different schemes in discovering gateways as we increase the number of requests. Our scheme achieves the lowest message overhead (less than 40000 bits for 1000 requests) when compared to the other schemes. This is mainly due to the fact that LAGAD determines efficiently the location-aided advertisement zone. The reactive approach has the highest bandwidth usage that increases from 2,157,084 bits to 21,371,868 bits, for a number of gateway requests ranging from 100 to 1000 requests. This results describes the increase in the number of requests and the bandwidth usage incurred by each request. The proactive approach and the hybrid approach with 5 hops achieve higher bandwidth usage than our LAGAD. The bandwidth usage of the latter is almost stable with the increase in the number of requests. This stability is mainly because the bandwidth usage in these approaches depends on the number of gateways in the VANet and their frequency of advertisement. As a consequence, the increase in the number of requests does not greatly affect the bandwidth usage of the proactive and the hybrid approach with a large advertisement zone. We infer also that a hybrid approach with a large advertisement zone has almost the same behavior as a proactive approach. Hybrid approaches with 1 and 3 hops respectively, achieve a higher bandwidth usage than our LAGAD. Their bandwidth usage increases slightly with the increase in the number of requests. This increase in the number of requests is due to the bandwidth used for the propagation of the gateway requests to the advertisement zones of gateways. Thus, we infer that hybrid approaches are better in terms of bandwidth usage than the reactive approach. Moreover, we infer that a hybrid approach with a large advertisement zone has almost the same high bandwidth usage than a proactive approach.

Our LAGAD outperforms reactive and proactive approaches in terms of bandwidth usage; and, it also outperforms the hybrid approaches with fixed advertisement zones. This proves that the location-aided advertisement technique in our hybrid gateway discovery protocol LAGAD succeeds in determining the efficient advertisement zone of gateways and bypasses all the drawbacks of existing approaches in large-scale setting. In addition, the caching technique of gateway information during the reply propagation phase used in our LAGAD protocol helps to maintain low bandwidth usage in our LAGAD. Figure 3.8(c) illustrates the average bandwidth usage per one successful query when we increase the number of gateway requests. For the proactive approach, the average bandwidth usage per one successful query is equal to zero when the number of requests increases from 100 to 1000. This means that all bandwidth generated in the proactive approach (as depicted in Figure 3.8(b)) for the discovery of gateways is incurred from the advertisement messages. On the other hand, the reactive approach has the highest average bandwidth usage per one successful query, when we increase the number of gateway requests. This is because of the fact that in the reactive approach there is no gateway advertisement. For hybrid approaches, the average bandwidth usage per one successful query increases with the decrease of the advertisement zone. This is due to the smaller the advertisement zone is, the less advertisement messages are generated. Our LAGAD has the lowest average bandwidth usage per one successful query when we increase the number of gateway requests in the hybrid approach. This proves that our LAGAD succeeds in determining its advertisement zone so that less request gateway discovery messages are propagated until they reach the adaptive advertisement zone.

In Figures 3.8(f), 3.8(g) and 3.8(h), we discussed respectively the average delivery rate, the average end to end delay and the average duplicates of data packets when gateway clients are involved in sending 10 data packets to their discovered gateways. Each data packet is 512 bytes and sent at 1 second intervals. Figure 3.8(f) depicts the average data packets delivery rate for different schemes of delivering data packets from source clients to the gateway, when the number of gateway requests increased from 100 to 1000. Our proposed LAGAD scheme achieves high data packets' delivery rates (more than 90%), when the number of gateway requests is increased. This is mainly due to the efficient and robust routing protocol (the CLA-S [40]) used for the communication between source vehicles and gateways. In this latter, routes are not established in a hop by hop basis and communication packets are retransmitted by all the nodes inside a predetermined forwarding area. Thus, multiple geographic paths are provided to relay data from the source to the destination, which increases the delivery ratio of LAGAD and permits duplicate data packets as depicted in Figure 3.8(h).

We noticed from Figure 3.8(f) that the reactive approach achieves an average delivery rate (in the order of 60%) when the number of gateway requests increased. The proactive approach has the lowest delivery rate (less than 20%). Hybrid approaches with different fixed advertisement zones also have low delivery rates (less than 30%). This is mainly due to the fact that in these protocols unicast is used to relay data from a source to the destination gateway. The reactive approach has higher delivery ratio than the proactive and hybrid approaches with fixed advertisement zones because the routing path, used to relay data packets from the sources to gateways, is more up to date than the routing paths used in proactive and classic hybrid approaches. However, due to the high mobility of vehicles in VANets, routing paths in the reactive approach cannot stay relevant when there are many data packets to be transmitted from the source to the gateway. The proactive approach has the lowest data delivery ratio because proactive routing paths cannot be maintained up to date in a highly mobile network. The classic hybrid approaches have low delivery rates of data packets because in the advertisement zone the routing path may not be up to date. The larger the advertisement zone is in the classic hybrid approaches, the lower the data packet delivery ratio is.

Figure 3.8(g) describes the average end to end delay of data packets. We notice that our LAGAD and the reactive approach achieve low data packets end to end delay (in the order of milliseconds). However, the end to end data packet delivery delay in the proactive and the classic hybrid approaches is high. This is mainly due to the delay related to the maintenance of the routing path in these approaches. For the packet order metric, we add timestamps containing the sending time of every data packet at the source vehicle. Then we verify at the destination whether or not the data packets are received in the same order as they were sent. Our LAGAD succeeded in maintaining the order of the sent data packets as opposed to the other approaches that failed in many cases to maintain the packet order.

We conclude that in a Manhattan traffic scenario, our scheme achieves quite a high success rate, and the lowest bandwidth usage with an acceptable response time when compared to proactive, reactive and hybrid schemes with fixed advertisement zones in a large-scale setting. In addition, our proposed LAGAD has good performance when relaying data packets from sources to destination gateways.

3.6 Summary

In this chapter, we presented a scalable hybrid adaptive Location-Aided Gateway Advertisement and Discovery protocol. A comparative study of LAGAD's performance and other gateway discovery approaches in a Manhattan traffic model has shown that LAGAD, the proactive protocol and the hybrid protocol (with 5 hops), achieved similar high success rates of gateway queries and low response times of gateway queries. With respect to message overhead and bandwidth usage, LAGAD has demonstrated a considerably lower overhead than the proactive, reactive and classic hybrid approaches, which permit the scalability of LAGAD.

Moreover, LAGAD has shown good performance when relaying data packets from sources to destination gateways. In fact, we showed that LAGAD has the highest delivery ratio of data packets with low end to end delays and permits duplicate and ordered data packets reception at the destination gateway. These metrics are relevant to many applications such as video/voice streaming.

Finally, we would like to state that our location-aided gateway advertisement mechanism could be applied to any existing hybrid gateway discovery protocol in wireless networks. The main weakness of protocols in hybrid approaches is to determine an efficient radius of the gateway advertisement zone. Thus, instead of using a fixed advertisement zone, like existing hybrid gateway discovery protocols, we believe that the use of our location aided advertisement mechanism could improve the performance of these protocols.

Chapter 4

Infrastructure based Vehicular Service Discovery Protocol: VSDP

4.1 Introduction

The number of applications in VANets is increasing rapidly [82, 19, 30, 54]. Two major types of vehicular networks applications are commonly known: *(i) road safety* applications and *(ii) driving comfort* applications. The former applications deal with the enhancement of vehicle safety on roads, while the latter ones help provide services, such as entertainment, voice or video streaming, or localization services, to drivers and passengers in their vehicles. Guaranteeing safety on the roads is one of the main objectives of vehicular networks. As a result, it is very important that the existence of convenience services do not decrease the performance of safety services. In fact, the dissemination of safety services or the discovery of convenience applications require the communication between service providers and service requesters through a limited shared bandwidth. Any solution to the service discovery problem in vehicular networks is subject to this limitation. A good set of communication protocols is the key to the efficient use of the vehicular network [15]. Future cars will be comprised of various sub-systems that would require efficient communication in order to exchange information and discover services. Vehicles in the future will include computing facilities and will need to communicate with other vehicles and with the fixed network. In the following, first we discuss our motivation and challenges. Second, we explain our contribution and research challenges. Finally, we present the chapter organization.

4.1.1 Motivation and Research Challenges

Service discovery is a crucial challenge in large-scale vehicular networks. In the literature, many service discovery protocols are proposed. They are classified in three categories: *(i)* Push-based discovery approach [18, 29, 58, 79, 80, 13, 83, 11], *(ii)* Pull-based discovery approach [69, 78, 25], and *(iii)* Hybrid discovery approach [10, 26, 36, 45, 71, 50]. To the best of our knowledge, none of the existing service discovery protocols were designed with the particularities of vehicular networks in mind. This will enable one to design an efficient and scalable service discovery protocol for such a special type of network. The characteristics of vehicular networks have motivated us to design an efficient and scalable service discovery protocol that considers the properties of such network.

We define a large-scale vehicular network as a network where the number of service requests is high. We choose to implement our service discovery protocol in an infrastructure-based vehicular network for two main reasons. First, an efficiently managed communication between vehicles through the wireless roadside backbone in large-scale congested networks increases the bandwidth usage of the vehicular network, which increases the number of satisfied service requests. Second, the wireless backbone increases the bandwidth usage in an intermittent large-scale vehicular network where vehicles could loose easily their connectivity because they are not in the range of each other. Thus, a good placement of Roadside Routers in an intermittent vehicular network allows the connectivity between the different vehicles and increases the number of satisfied requests. Consequently, the bandwidth usage is increased.

4.1.2 Contribution and Objectives

In this chapter, we propose a novel service discovery technique dedicated for large-scale vehicular networks. Our approach integrates the discovery information into the routing layer with the routing messages. Based on an adaptively adjusted and predictable advertisement zone size of service provider vehicles, we have combined proactive and reactive service discovery strategies to come up with an efficient and scalable hybrid adaptive service discovery protocol for vehicular networks. We use also the channel diversity feature in order to decrease the congestion on single channels and enlarge the usage of the radio spectrum. Our protocol targets the two main types of convenience services: *(i) Fixed services* have a fixed location. Their position does not change over time. Examples of services in this category could be: restaurants with their menus, gas stations with their price lists, parkings with available spots, etc; *(ii) Moving services* are the services provided by the vehicles on roads. The locations of these services depend from the locations of the moving vehicle. Example of services in this category could be: music sharing, game sharing, file sharing, etc. Our protocol is suitable for both types of services (fixed and moving services) because the discovery reply messages and data communication packets between service providers and service requesters use an efficient packet based routing protocol for highly mobile networks (CLA-S [40]) as we will explain in our system model. Thus, even if service providers or intermediate vehicles are moving, messages will eventually reach their destinations in highly mobile vehicular networks.

4.1.3 Chapter Organization

The remainder of this chapter is organized as follows. Section 4.2 describes our system model. Section 4.3 contains a detailed description of our service discovery algorithm, the Vehicular Service Discovery Protocol (VSDP). Section 4.4 presents the proof of correctness of our protocol. Section 4.5 discusses the performance analysis of VSDP. Section 4.6 reports on the simulation experiments we have conducted in order to evaluate the performance of our algorithm. Section 4.7 concludes the chapter.

4.2 System Model

Our system model comprises two types of components: Roadside Routers (RRs) and Road Vehicles (RVs) that are described below:

1. ***Roadside Routers (RRs):*** are deployed in the congested areas of the VANet. They are connected to each other and form an 802.11-based wireless Roadside Backbone (RB) in every congested area. They are fixed or have low mobility, and they have routing capabilities. Roadside Routers are equipped with two radio interfaces and use diverse channels. RRs capacities and capabilities are improved when an efficient utilization of channels is taken into consideration. For packet routing between RRs, we implement an enhanced version of the SHARP routing protocol [67]. SHARP is a hybrid routing protocol for ad hoc networks. Our enhanced version of SHARP consists of using multiple interfaces and multiple channels. SHARP combines proactive and reactive routing protocols. The SHARP Proactive Routing (SPR) protocol is an efficient protocol that uses techniques from different routing algorithms such as Destination Sequenced Distance Vector DSDV [65] and Temporally Ordered Routing Algorithm TORA [62]. In each proactive zone, the SPR protocol builds and maintains routes to a single destination. This permits to SPR to use efficient mechanisms for building and maintaining routes with low overhead and offers good properties like route overhead prediction. We modify the original version of SPR by considering channel diversity during the routing process. In the proactive zone of the routing protocol, route discovery is node-specific and routes are maintained only to the central RR correspondent to the vehicle service provider. In the reactive zone, SHARP uses the AODV routing protocol [64]. In our case, we use a multi-channels and multi-interfaces version of AODV. Multihop route to the desired service provider should be selected carefully. In fact, in a multihop route, successive hops may not use different channels, consequently channel diversity is not fully utilized. Most of the existing routing protocols use the minimum number of hops as a unique metric in order to select a route to the destination such as AODV [64] and

DSR [43]. These routing protocols do not take into consideration channel diversity, they do not distinguish between routes that use few channels and routes that use many channels. These latter could have more hops but they provide better aggregate throughput because they use different channels on each hop.

2. ***Road Vehicles (RVs):*** are the cars circulating along the roads and equipped with various devices. Cars could be equipped with one wireless interface. Cars that are not equipped with a wireless interface cannot participate in the execution of our protocol, thus they will not affect our algorithm. For the participating cars, the interface is dedicated for the ad hoc communication with neighboring vehicles and for the communication with Roadside Routers through their ad hoc wireless interface. Wireless communication between vehicles and Roadside Routers is based on the IEEE 802.11 protocol. Road Vehicles could be connected to each other either directly, or through the wireless Roadside Backbone. The connection to the wireless Roadside Backbone could be performed in multi-hop through intermediate vehicles. Vehicles are characterized by their high mobility and their increased density. The particular mobility model in VANets is an important property that distinguishes it from MANets. In fact in VANets, nodes' mobility is not completely random, it is restricted by the roads. For the communication between vehicles and the communication between vehicles and RRs, we use the Connectionless Approach for Street CLA-S [40] as routing protocol. The latter is a packet based routing protocol. It is robust and well suited for highly mobile vehicular networks. In CLA-S [40], routes are established in hop by hop basis toward a predetermined location. Thus, it is based on geographical information.

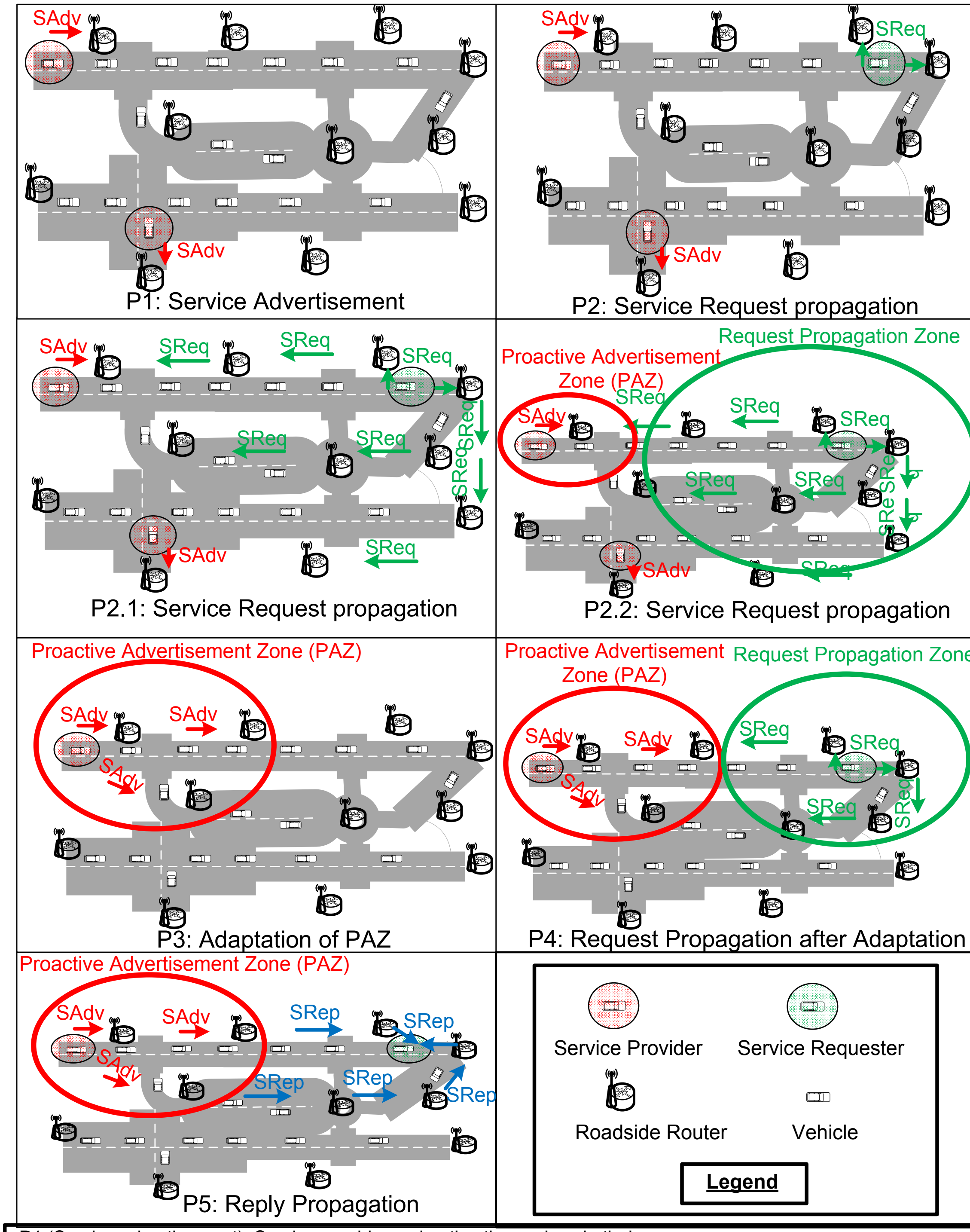
In our system model, we suppose that vehicles move on a two dimensional plane. We assume a rectangular topology of surface $a \times b$, where a and b are the length and the width of the VANet respectively. We consider a number of nRR Roadside Routers that are uniformly distributed along the vehicular network. We model the arrival process to the vehicular network as a poisson process with an arrival rate (λ). The density of the network is $Density$ and it provides the number of vehicles per meter square in the VANet. The density of a road section is equal to the density of the VANet. We assume that a road section has a fixed length and a fixed surface. The average speed of a vehicle is S. We assume the random variable M that defines the number of vehicles in a road section of length y. The probability that there are n vehicles in a road section of length (y) is given by:

$$P[M(y) = n] = \frac{(\frac{y}{S}\lambda)^n}{n!} e^{-\lambda \frac{y}{S}} \tag{4.1}$$

We assume the service level D according to the traffic flow state for different densities in [55], where the network is uncongested but borders are unstable. Accordingly, we assume that the average network density in vehicles is 120 vehicles/Km^2 and vehicles are moving with an average speed S of 70km/h (≈20m/s). Consequently, we deduce the arrival rate (λ) for our simulations ($\lambda = \lambda(y) = \frac{\overline{M(y)} S b}{y}$), where $\overline{M(y)} = \sum_{n=1}^{\infty} nP[M(y) = n]$.

4.3 Description of The Vehicular Service Discovery Protocol: VSDP

In this section, we present the design of our proposed hybrid adaptive Vehicular Service Discovery Protocol (VSDP) for large-scale vehicular networks (VANets). An illustrative example is presented in Figure 4.1. VSDP is an infrastructure based discovery protocol integrated with an efficient hybrid adaptive routing protocol. The fact of integrating VSDP with the routing protocol results in a lightweight discovery protocol that largely decreases the overhead caused by the unnecessary flooding of the network. This is because, when the required service is found, its routing information is piggybacked in the service reply message. A summary of the main characteristics of VSDP is provided in Figure 4.2. In this Figure, service providers advertise themselves through their corresponding Roadside Routers. We express the advertisement zone size in terms of number of hops. Service requesters send their requests



.P1 (Service advertisement): Service providers advertise themselves in their range.
.P2 (Service request propagation): A service requester initiates a request message and sends in its range.
  .P2.1: The request message is propagated by roadside routers and vehicles.
  .P2.2: The request message reaches the PAZ of the requested service provider.
.P3 (Adaptation of PAZ): The SP adapts its PAZ for future requests.
.P4 (Request propagation after adaptation): The propagation zone of future request messages of the same SP is reduced after the adaptation of the PAZ.
.P5 (Reply propagation): Service reply messages are sent to the requester from road components on the border of the PAZ.

Figure 4.1: VSDP Execution

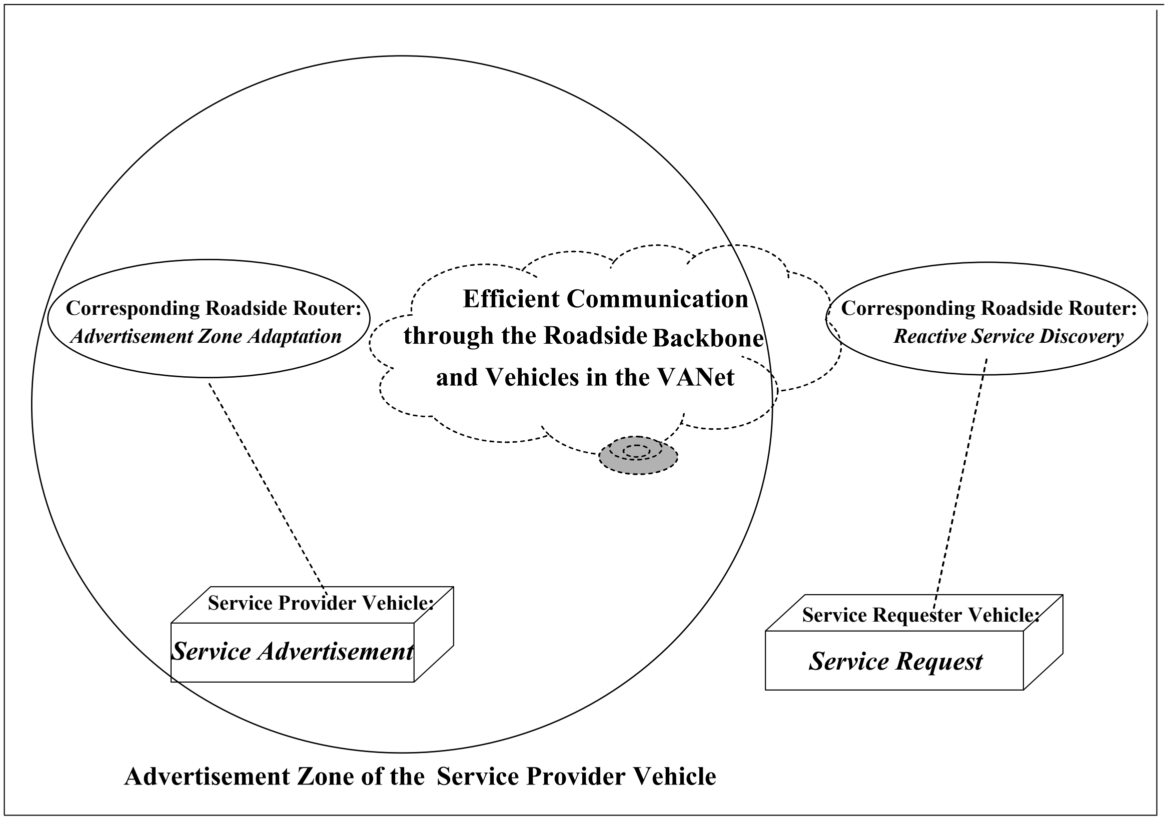
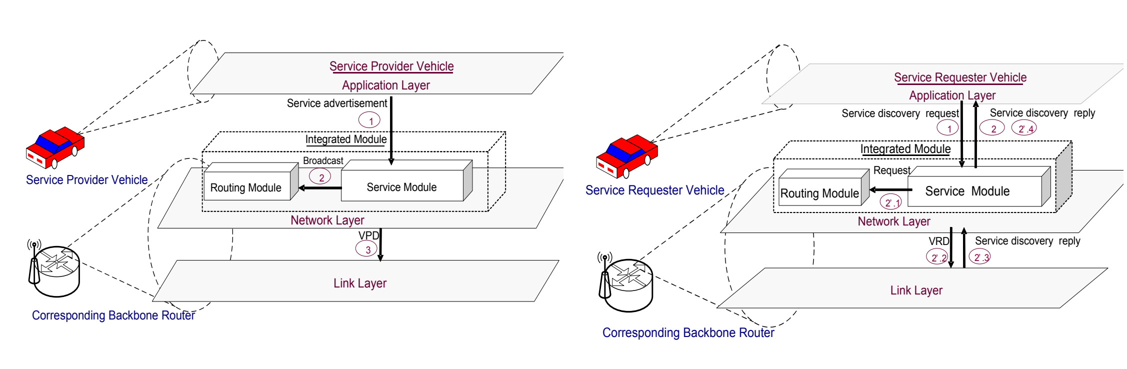


Figure 4.2: Overview of the Vehicular Service Discovery Protocol: VSDP

reactively through their corresponding Roadside Routers. The main feature of VSDP is the ability of its service provider vehicular nodes to adjust the size of their advertisement zone dynamically and autonomically depending on network conditions such as network congestion, the mobility rate, the popularity of the service provider, etc. VSDP efficiently determines the advertisement zone's hop size of the service provider and combines reactive and proactive discovery strategies in order to decrease the traffic in the Roadside Routers backbone. In what follow, we shall explain how we implemented the main features of our VSDP which are : *(i)* Network layer-based VSDP; *(ii)* Multiple-interfaces and multiple-channels VSDP; and *(iii)* scalable hybrid adaptive VSDP.



(a) Vehicular Proactive Discovery (b) Vehicular Reactive Discovery

Figure 4.3: Framework of the network layer based-VSDP

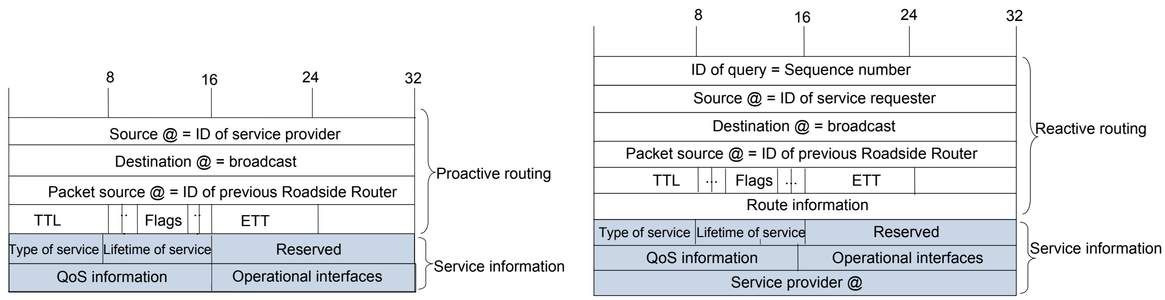
4.3.1 Network-Layer-Based VSDP

Our proposed protocol consists of *lightweight advertisement* and *discovery techniques*. Its purpose is to save the network resources by finding a service provider as well as its routing information. In large-scale wireless vehicular networks, redundant flooding due to service advertisement and service requests can engender network congestion and drastically affect the limited resources in wireless vehicular networks. By piggybacking the service discovery in the routing information messages, an efficient network-layer-based service discovery protocol can be implemented. VSDP uses a hybrid service discovery model that permits each vehicle to perform proactive service discovery if it is inside the proactive discovery zone of the specific service provider and a reactive service discovery outside that zone. In order to do this, we need to integrate our service discovery protocol with an efficient hybrid routing protocol. In the literature, hybrid routing protocols have been considered more efficient than simple proactive or reactive protocols [72, 34]. For this reason, we believe that the integration of a service discovery protocol in a hybrid routing protocol would outperform the simple integration in proactive or reactive routing protocols. As we explained earlier in the system model, we integrate our service discovery protocol with the hybrid routing protocol *SHARP* [68].

Figure 4.3 provides a simplified architecture of the framework resulting from the integration of the service discovery module and the routing module into the network layer. It provides an overview of the collaboration between the routing module and the service module. Figure 4.3 illustrates the process of our service discovery protocol. When an application program installed in a vehicle device wants to find an appropriate service, a service discovery request is sent to the service discovery module of the nearest Roadside Router (RR) in the Roadside Backbone (RB). The RR verifies whether or not the requested service exists in the service information table. If the requested service does not exist in the service information table, then the service discovery module requests that the routing module collaborates to find the requested service, by generating the appropriate route request packet and by passing the application request to the integrated module. It is this module that will generate the appropriate VRD request packet. A VRD request packet contains both service and routing information. Upon reception of a VRD request packet by a Roadside Router, the packet is decoded by the integration module. This enables the separation of the routing information from service information, which will be treated differently by the routing module and the service discovery module respectively.

Our VSDP protocol creates proactive zones around vehicles that provide the service. The size of the advertisement proactive zone depends on many features, such as the popularity of the service provided by the vehicles and the network congestion. Once a packet has been received by a peripheral Roadside Router node of the proactive zone, it is efficiently forwarded to the vehicle providing the service. VSDP maintains and efficiently spreads routes to the destination when there are many vehicle sources communicating with the service provider vehicle. This prevents the development of bottleneck zones in the network.

Figure 4.4(a) illustrates VPD (Vehicular Proactive Discovery) packets, which consist of the proactive routing (PR) packets that carry piggybacked service advertisement information. Figure 4.4(b) shows VRD (Vehicular Reactive Discovery) packets that consist of the reactive routing (RR) packets that carry piggybacked service discovery information. VPD packets are used by a service provider vehicle in order to advertise its service in the proactive discovery zone. Outside the proactive discovery zone, VRD packets are used to generate service dis-



(a) VPD Packet Format (b) VRD Packet Format

Figure 4.4: Packets Format

Service Provider	Service information
Service provider 1	Type
	Lifetime
	Other information
Service provider 2	. . .
Service provider ...	. . . .

(a) VSDP Service Table Format

Destination	Routing information
Roadside Router 1	List of Ids of Roadside Routers
Roadside Router 2	. . . .
Roadside Router ...	. .

(b) VSDP Routing Table Format

Figure 4.5: Tables Format

covery queries. The proactive discovery mechanism uses VSDP Proactive Discovery (VPD) packets. The format of a VPD packet is illustrated in Figure 4.4(a). The VPD packet contains two parts. One part contains information about the proactive routing (PR). The other part contains information about the service provider such as the type of the service, the lifetime of the service, the QoS of the service, how to interfere with this service (port number, parameters, etc.), and additional information can be put in the reserved space that contains the location of the service provider required for the CLA-S [40] routing protocol.

We assume that all vehicles have the same naming for service types. The service lifetime provides the period expected by the service to be available in the vehicular network. The QoS of the service is optional and it provides information about the performance and capacities of the service provider. The Proactive Routing (PR) part of the VPD contains the address of the service provider as a source address, the time-to-live (TTL) of the VPD packet that is initiated to the size of the proactive advertisement zone and the destination is a broadcast till the TTL equals 0. Upon a Roadside Router receives a VPD packet, it stores the essential information in the service and routing table illustrated respectively in Figures 4.5(a) and 4.5(b). Service information tables are consistent in every RR through the service advertisement process.

Figure 4.4(b) illustrates the message format of a VRD request packet. As for VPD packet, there are two parts that construct the VRD request packet. The first part contains information about the service requested, and the second part contains information about the routing. Initially for the first part, the fields lifetime and service provider address are initiated to null. Data that identifies the service that the user is requesting are stored in the service type field. The additional information field stores the location of the requesting vehicle that will be used by the CLA-S [40] routing protocol. The destination address field of the VRD request packet contains a broadcast. The VRD request is broadcast in all the channels. Every new VRD packet has a unique sequence number. A VRD request packet is retransmitted if its sequence number has been seen for the first time; in this case it is stored in the local discovery and routing tables.

The service reply packet contains the service information collected from the VRD request packet. After finding the location of the requested service, a service reply message is sent to the vehicle requesting the service through the CLA-S [40] routing protocol. The latter uses the location information of the requesting vehicle stored in the VRD request packet to send the reply message. The location of the service provider is added to the reply message. Data communication between the service provider and the service requester is done using the CLA-S [40] as routing protocol. The latter is suitable for highly mobile vehicular networks.

4.3.2 Multi-Channels and Multi-Interfaces VSDP

Since vehicular networks capacities and capabilities are enhanced through the usage of diverse channels, we choose to implement the multi-channels and multi-interfaces feature in our proposed protocol. This feature in handled at the Roadside Routers that are equipped with

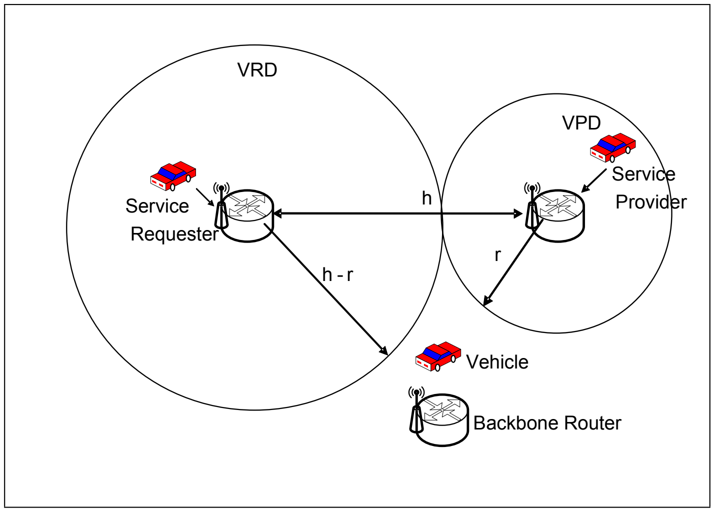


Figure 4.6: VRD and VPD coverage

at least two interfaces (interface1 and interface2). We set a different fixed channel on each interface of the same Roadside Router. We use two different channels in the vehicular network in order to guarantee its connectivity and to avoid channel switching costs. We set interface1 to the fixed channel1, and we set interface2 to the fixed channel2. The routing module of each Roadside Router determines which interface is used for the propagation of VSDP messages, so that sending and receiving of the discovery messages are performed simultaneously through the different interfaces.

4.3.3 VSDP Adaptation: Minimal Packet Overhead

Our VSDP protocol adapts efficiently and seamlessly to proactive, reactive and hybrid discovery strategies. This adaptation depends on the characteristics of the network and the requirements of a given application service. It aims at enhancing some performance metrics, such as service availability, network overhead or discovery latency. The adaptation ability of our service discovery protocol consists mainly of varying the amount of information shared in a proactive way in order to reduce the overhead in the wireless roadside backbone. The proactive zone employs a node-specific proactive discovery protocol, where routes are maintained only to the vehicle service provider. Each vehicle inside the advertisement zone is a member of the proactive zone for that vehicle service provider. All vehicles outside the provider's advertisement zone use reactive discovery in order to reach the vehicle service provider.

In our protocol, if the advertisement zone's size is 0 hops, then the discovery protocol is completely reactive. Conversely, if the advertisement zone's size is equal to the diameter of the vehicular network, then the discovery protocol is proactive. VSDP aims to determine the size of the proactive advertisement zone that would reduce the Roadside Routers traffic overhead in the vehicular network compared to pure proactive or pure reactive approaches. The strategy that we use in order to find the appropriate advertisement zone's size depends on the trade-off between the reactive and the proactive discovery components. The number of requesting vehicles does not affect the cost of the proactive discovery component, nor does the mobility rate affect the proactive discovery cost. However, the reactive discovery depends largely on the number of requesting vehicles and on the mobility rate. Reactive service discovery generates a higher aggregate cost when the mobility rate in the network increases.

Our proposed adaptation process consists of: (i) an *adjustment technique*; and (ii) a *prediction technique*. The adjustment technique uses an adjustment procedure as illustrated in Algorithm 4.3.1. The latter compares the variation of the proactive component overhead to the variation of the reactive component overhead, then if necessary, it adjusts accordingly the size of the advertisement zone. The prediction technique is shown in Algorithm 4.3.2, it uses a prediction procedure based on the expected number of requests in the next adaption period. It computes the expected variation of the proactive component overhead and the expected variation of the reactive component overhead. Then, if necessary, it predicts the advertisement zone size in the next adaptation period for a best tradeoff between the reactive and the proactive components. In Table 5.1, we describe the variables used in the Algorithms.

The adaptation process is triggered by the service provider every adaptation period. The adaptation period is usually larger than the advertisement period of the service provider. In our protocol, we run our simulation with different adaptation periods and we deduced that the performance of our protocol is good when the adaptation period is set at least to three

Table 4.1: Variables description

Variable Name	Description
P	service provider
R	service requester
RR	Roadside Router
ProactiveOverhead	stores the computed proactive overhead
ReactiveOverhead	stores the computed reactive overhead
ExpectedProactiveOverhead	stores the expected proactive overhead
ExpectedReactiveOverhead	stores the expected reactive overhead
$advP$	advertisement period of a service provider
$adaptP$	adaptation period
$NRR(RR_i, r_i)$	function that returns the number of Roadside Routers neighbors of the RR of vehicle i and located r_i hops from RR
$nRAdaptP$	the number of service requests during the adaptation period $AdaptP$
h_{PR}	number of hops separating the RR of the R from the RR of P
$advSize$	number of hops in the proactive advertisement zone of P
$ExpectednRAdaptP$	the expected number of service requests during the next adaptation period $AdaptP$
$Expectedh_{PR}$	the expected number of hops separating the RR of the R from the RR of P
$avgRR(1)$	average number of RRs in one hop
tr_0	RR maximum transmission range
a	the length of the vehicular network

Algorithm 4.3.1 THE ADJUSTMENT PROCEDURE

Action: $ProactiveOverhead = (1 + NRR(RR_P, advZone_P)) \times \frac{adaptP}{advP}$
{Number of advertisement messages sent by P and forwarded by its corresponding RR in the provider's advertisement zone during $adaptP$} {Number of request messages sent by every R and forwarded by their corresponding RR until the proactive advertisement zone of the current P during $adaptP$ (see Figure 4.6)}
Action: $ReactiveOverhead = \sum_{R=1}^{nRAdaptP}(1 + NRR(RR_R, (h_{PR} - advZone_P)))$
{The number of hops (h_{PR}) separating the RR of P from the RR of R is sent to P in a connection message $conn\_msg$ every time a service request message reaches the border of the advertisement zone of the requested P.}
Action: ProactiveVariance = ProactiveOverhead - PreviousProactiveOverhead;
Action: ReactiveVariance = PreviousReactiveOverhead - ReactiveOverhead;
{if the computed reduction in reactive component overheads larger than the computed augmentation in the proactive component overhead}
1: **if** (ReactiveVariance > ProactiveVariance) **then**
Action: $advZone = advZone + \frac{ReactiveVariance - ProactiveVariance}{avgRR(1)}$
{The service provider vehicle decides to increase the size of its advertisement zone in terms of hops as follow } {if the computed decrease in the proactive component overhead is larger than the computed increase in the reactive component overhead}
2: **else if** (ProactiveVariance > ReactiveVariance) **then**
Action: $advZone = advZone - \frac{ProactiveVariance - ReactiveVariance}{avgRR(1)}$
{The service provider vehicle decides to decrease size of its advertisement zone in terms of hops}
Action: PreviousProactiveOverhead = ProactiveOverhead;
Action: PreviousReactiveOverhead = ReactiveOverhead;
3: **end if**

times the advertisement period. Thus, we decided to set the adaptation period of every service provider P to three times its advertisement period ($adaptP = 3 \times AdvP$). Initially, the size of the advertisement zone of every service provider is equal to 1. Every adaptation period, the service provider uses consecutively the procedures illustrated in Algorithms 4.3.1 and 4.3.2 in order to decide whether to increase, decrease or not to alter the advertisement zone size in terms of number of hops. In the following, we formally describe the proposed Vehicular Service Discovery Protocol (VSDP) by means of pseudo-code. The algorithm comprises two main parts: The first part is running on the Vehicle V_i as illustrated in Algorithm 4.3.3; and

Algorithm 4.3.2 THE PREDICTION PROCEDURE

Action: $ExpectedProactiveOverhead = (1 + NRR(RR_P, advZone_P)) \times \frac{adaptP}{advP}$
{Expected number of advertisement messages sent by P and forwarded by its corresponding RR in the provider's advertisement zone, during the next period $adaptP$.}
Action: $ExpectednRAdaptP = adaptP * \frac{1}{E(x_b)}$ (see equation 4.6)
{Expected number of request messages sent by every R and forwarded by their corresponding RR until the proactive advertisement zone of the current P during the next period.}
Action: $Expectedh_{PR} = \frac{0.521 \times a}{tr_0}$ (see equation 4.14)
Action: $ExpectedReactiveOverhead = \sum_{R=1}^{ExpectednRAdaptP}(1 + NRR(RR_R, (Expectedh_{PR} - advZone_P)))$
Action: ExpectedProactiveVariance = ExpectedProactiveOverhead - ProactiveOverhead;
Action: ExpectedReactiveVariance = ReactiveOverhead - ExpectedReactiveOverhead;
{if the expected reduction in reactive component overhead is larger than the expected augmentation in the proactive component overhead}
1: **if** (ExpectedReactiveVariance > ExpectedProactiveVariance) **then**
Action: $advZone = advZone + \frac{ExpectedReactiveVariance - ExpectedProactiveVariance}{avgRR(1)}$
{The service provider vehicle decides to increase the size of its advertisement zone in terms of hops}
{if the Expected decrease in the proactive component overhead is larger than the Expected increase in the reactive component overhead}
2: **else if** (ExpectedProactiveVariance > ExpectedReactiveVariance) **then**
Action: $advZone = advZone - \frac{ExpectedProactiveVariance - ExpectedReactiveVariance}{avgRR(1)}$
{The service provider vehicle decides to decrease size of its advertisement zone in terms of hops}
3: **end if**

the second part is running on the Roadside Router RR_i as illustrated in Algorithm 4.3.4.

Algorithm 4.3.3 PROTOCOL AT THE VEHICLE VI

1: **Case A.** V_i has a service to advertise:
2: **Step A0.** Periodically (Every $\Delta_{adaptation}$): Determine the number of hops for advertisement ($nhops$) as in Algorithms 4.3.1 and 4.3.2.
3: **Step A1.** Periodically (Every $\Delta_{advertisement}$): Send the advertisement message in $nhops$
4: **Case B.** V_k needs to find a service s:
5: **while Step B0.** N<3 ***or*** $SRep$ message is NOT received **do**
6: **Step B0.0.** V_k sends a service request $SReq$ to its correspondent Roadside Router (RR) directly or through multihops
7: **Step B0.1.** Wait for a predetermined period of time.
8: **end while**
9: **if Step B1.** ($SRep$ message received) **then**
10: **Step B1.0.** GOTO Step C0.
11: ***else***
12: **Step B1.2.** Exit: Service not found
13: **end if**
14: **Case C.** V_k receives a service reply :
15: **Step C0.** Store service provider information.
16: **Step C1.** Extract service provider routing information.
17: **Step C2.** Establish the connection with the provider through CLA-S[40].

4.4 Proof of Correctness of VSDP

The proof of correctness of our proposed Vehicular Service Discovery Protocol (VSDP) can be done in three steps. First, we need to prove that the discovery protocol is accurate by showing that the returned result matches effectively the user's request as in Lemma 4.4.1. Second, we have to show that the discovery protocol is complete by showing that the protocol finds all the services that satisfy the query in a finite time as in Lemma 4.4.2. Third, we need

Algorithm 4.3.4 PROTOCOL AT THE ROADSIDE ROUTER RRI

<u>**Case D.**</u> receive a service advertisement message $SAdv\_msg$ from a vehicle V_i:
Step D0. The Service Module updates the service information table.
Step D1. The Routing Module updates routing table.
if Step D2.($nhops > 0$) **then**
    Step D2.0. $nhops = nhops - 1$.
    Step D2.1. The Integrated Module generates the appropriate VPD message.
    Step D2.2. Forward the VPD message.
end if
<u>**Case E.**</u> receive a VPD message from a Roadside Router RR_j:
Step E0. Decode the packet by separating the routing information from service information.
Step E1. The Service Module updates the service information table.
Step E2. The Routing Module updates the routing table.
if Step E3. (nhops > 0) **then**
    Step E3.0. $nhops = nhops - 1$.
    Step E3.1. The Integrated Module forwards the VPD message.
end if
<u>**Case F.**</u> receives a request message $SReq\_msg$ from V_k:
Step F0. The Service Module checks the service description.
if Step F1. service exists in service information table **then**
    Step F1.0. return a service reply message $SRep$ with routing information from the Routing Module to V_k.
else
    Step F1.1. The Service Module solicits the Routing Module to collaborate in finding the requested service.
    Step F1.2. The Routing Module generates the appropriate route request $RReq$ packet.
    Step F1.3. The Integrated Module generates a Vehicular Reactive Discovery VRD packet.
    Step F1.4. Propagate the VRD packet to the neighboring Roadside Routers.
end if
<u>**Case G.**</u> receive a VRD request message from Roadside Router RR_j:
Step G0. decode the packet by separating the routing information from service information.
Step G1. Send service request $SReq$ to Service Module.
if Step G2. service exists in service information table **then**
    Step G2.0. send a service reply message $SRep$ with routing information to RR_j through CLA-S[40].
else
    Step G2.1. ***GOTO*** Step F1.1.
end if
<u>**Case H.**</u> receives a service reply message $SRep$ from Roadside Router RR_j:
if Step H0. V is the service requester **then**
    Step H0.0. service found.
else
    Step H0.1. forward the reply message to its destination using CLA-S[40].
end if

to prove that our discovery protocol terminates in a finite time and that there are no deadlocks as in Lemma 4.4.3.

Lemma 4.4.1. ***(VSDP Accuracy)*** *The Vehicular Service Discovery Protocol (VSDP) finds a service that satisfies the requesting vehicle in a finite time if the requested service is provided in the vehicular network and the network is connected.*

Proof: Let us assume that the Vehicular Service Discovery Protocol cannot find the requested service to the requesting vehicle V_j and will not receive a reply message $SRep$ even though there is a service that satisfies the requester in the vehicular network and the network is connected. The requesting vehicle sends a service request $SReq$ to its corresponding Roadside

Router *RR* as in *Step B0.0*. The Service Module of *RR* checks the service information table and if the service exists in its table, it returns directly a service reply $SRep$ message to the requesting vehicle as in *Step F1.0*. Otherwise, a Vehicular Reactive Discovery process is initiated and a *VRD* packet is sent to the neighboring Roadside Routers as in *Step F1.1 to Step F1.4*. The discovery search process continues as in *Step G*. If the requested service exists in the vehicular network, then a service reply $SRep$ is generated and is sent back to the Requester as in *Step G2.0*. Thus, the requesting vehicle will receive a service reply in a finite time. Therefore, V_j will eventually find the requested service in a finite time. This is a contradiction. Therefore, our assumption is false and the lemma is proved.

Lemma 4.4.2. ***(VSDP Completeness)*** *Assume that there are many services in the search area of a requester that satisfy the request, the requester will receive replies from all service providers that satisfy the request in a finite time.*

Proof: Let us assume that all the Roadside Routers in the search area of the requester V_j are connected. Since all the Roadside Routers in the search area of V_j are connected, the $SReq$ message will be received first by the corresponding Roadside Router. If the service requested exists in the service information table, then the Service Module will send a reply to the requester for each service provider as in *Step F1.0*. Second, if the service is not found in a proactive way, then the discovery request will be propagated to the neighboring Roadside Routers that will send back a reply if they have the requested service or they continue to propagate the request as in *Step G2*. Therefore, knowing that Roadside Routers in the vehicular network are connected, the requesting vehicle will eventually receive reply messages from all service providers in a finite time.

Lemma 4.4.3. ***(VSDP Correctness)*** *Our Vehicular Service Discovery Protocol ($VSDP$) will eventually terminate in a finite time even if there is no provider that satisfies the driver's or passenger's query.*

Proof: Let us prove that our VSDP terminates the discovery process in a finite time. The loop of the Vehicular Service Discovery Protocol always returns a service reply message $SRep$

if there is at least one service that satisfies the request. If there is no service that satisfies the request after three attempts to find the service, *Step B1.2* allows the process to be terminated in a finite time. Hence, either the loop finds a service and *Step B1.0* is performed or the protocol terminates in *Step B1.2* in a finite time.

4.5 VSDP Performance Analysis

In this section, we discuss the performance and the message complexity of our VSDP protocol. Then we provide the cost functions of our service discovery technique.

4.5.1 Adaptive Backbone-Based Discovery Performance

Backbone-based service discovery in VANets should perform well in large-scale congested, or intermittent vehicular networks, because the communication is done through the wireless roadside backbone, which increases the bandwidth usage. Since the discovery messages are exchanged over the wireless roadside backbone, our algorithm reduces the network traffic considerably compared to infrastructure-less discovery protocols. Besides, in highly dynamic vehicular networks, the communication through the wireless roadside backbone decreases the number of dropped queries due to the movement of service providers and service requesters vehicles. This is because the usage of the wireless backbone maintains a certain level of stability during the discovery process. In fact, in our protocol, Roadside Routers are aware of the mobility of vehicles and can route the required discovery messages according to vehicles' mobility.

Our Vehicular Service Discovery Protocol ($VSDP$) finds the appropriate hybrid protocol with a low overhead by adapting its Advertisement Zone size and determining the best trade-off between the costly predictable proactive discovery protocol (VPD) and the variable cost of the reactive discovery protocol (VRD). In fact, in our protocol the size of the *Advertisement Zone* of each service provider is not fixed. We used an adaptation mechanism based on an adjustment procedure (Algorithm 4.3.1) and a prediction procedure (Algorithm 4.3.2) that try to find the appropriate hybrid protocol with low overhead by determining the best trade-off between the costly predictable proactive discovery protocol (VPD) and the variable cost of the reactive discovery protocol (VRD). $VSDP$ aims to decrease the total cost by dynamically determining the appropriate advertisement zone's size of the proactive discovery zone as in *Step A0*. The service provider vehicle decides to increase the value of the advertisement zone's size, if the reduction in the overhead for the reactive component is larger than the augmentation in overhead for the proactive component. Similarly, if a decrease in overhead for the proactive component is larger than an increase in overhead for the reactive component, then the value of the advertisement zone's size is decreased. If none of the above conditions apply, the advertisement zone's size is not altered. Thus, for every *adaptP* period, every service provider adapts its *Advertisement Zone size* for the aim of guaranteeing less message complexity during all the time. The computation of the new size of the advertisement zone is presented in Algorithms 4.3.1 and 4.3.2.

It is also important to know that the number of service providers and the number of service requests varies during the simulation period which means that a good advertisement zone's size at time $t1$ could not be the good one at time $t2$ of the simulation time. That is why, our adaptation process is performed periodically during all the simulation time in order to guarantee good performance at all the simulation time.

4.5.2 Message Complexity

As we mentioned earlier, VSDP is well suited for large-scale highly congested networks where the implementation of our protocol helps increase the success rate of service queries and permits an efficient bandwidth usage. It is also well suited for intermittent vehicular networks where the implementation of our protocol permits the connectivity between service providers and service requester. Thus, in both scenarios, we assume that vehicles communicate between each other through the wireless backbone and that vehicles have RRs in their ranges. However,

Table 4.2: Comparison parameters

Parameter description	Parameter value
Vehicles number	N nodes
RRs number	nRR
Period	p
Average network diameter	D
Vehicle transmission range	t_r
Time period	1 unit of time
Rate of service requests	k queries per vehicle per unit of time
Total number of requests	N / k (per unit of time)
Computation time period	t
Percentage of RRs inside the advertisement zone	α
Advertisement message interval	λ

it is worthy to note that our protocol is also suitable for the kind of scenarios where vehicles are not connected directly to RRs but use multihops to reach a RR.

In the following we compute the message complexity of VSDP and its cost functions using the previously mentioned well suited scenarios. Thus, messages are exchanged through the wireless roadside backbone. Consequently, the message complexity does not depend on the number of vehicles but on the number of Roadside Routers, which is usually less than the vehicles' number. Besides service information is stored in Roadside Routers inside the advertisement zone of a service provider, which can decrease the number of potential forwarded request messages. If the requested service is stored in the service information table of a Roadside Router, both time and bandwidth are saved for the service search and the reply propagation.

The scalability of our protocol is established when we increase the number of service requests. Assuming N is the number of vehicles and nRR is the number of Roadside Routers, the number of packets for one query is $O(nRR)$. In the best case where Roadside Routers are efficiently distributed, $nRR = log(N)$, thus the number of packets for one query is $O(log(N))$. In the other cases, $nRR = N^c$ where $0 < c < 1$, thus the number of packets is $O(N^c)$. Consequently, the message complexity for VSDP is expected to be in the best case $O(log(N))$ and $O(N^c)$ in the other cases. In the following, we compare the message complexity of VSDP to the SLP protocol.

4.5.2.1 Parameters and Assumptions

Table 4.2 lists the main parameters used for the comparison between VSDP, and SLP [33]. We assume an infrastructure-based vehicular network with a backbone of wireless Roadside Routers distributed uniformly over the network. Vehicles follow a traffic mobility model restricted by the roads. We assume that all Roadside Routers have the same transmission range and all the vehicles have also the same transmission range. We assume the service discovery packet sizes illustrated in Figures 4.4(a) and 4.4(b).

4.5.2.2 Cost for the Service Location Protocol SLP

SLP [33] consists of three types of agents: the service agent, appears on behalf of the service provider, the user agent, appears on behalf of the vehicle that wants to find a specific service, and directory agents that appear on behalf of service directories to manage the different services. The service agent subscribes its service with the directory agent and a user agent looks for a specific service request by interacting with the directory agent. SLP on VANets allows a user agent to interact directly with a service agent. User agents multicast their service requests. Service agents that satisfy their request, unicast their reply messages with a service provider's information and location. Service requests that failed to receive a reply will be retransmitted after a predefined waiting period set to 2 seconds. The waiting interval is increased exponentially until it reaches a predefined maximum waiting period set to 15 seconds.

A user agent that receives a first reply may use this first reply or could wait until it receives more replies. This is controlled in the algorithm by using a boolean variable. A list of previous responders is stored in a service request. Initially, this list is empty. A service agent that sees his address in the list of previous responders will not reply to the request. The request message is retransmitted until no more new responders are found, or a maximum retransmission period is reached, or a packet size limitation is met.

The scalable Bcast routing protocol developed by Kunz[48] has been used for multicasting the service requests. Bcast does not construct, nor maintain any distribution structure. In

Bcast, nodes exchange periodic control messages such that every node knows its 2 hop neighbors. A node A that receives a broadcast message from another node B, rebroadcasts it after a random delay if A has neighbors that will not be reached from B's broadcast. A modification to Bcast has been used for unicasting the service replies, where every node is considered as a multicast group. In the following we compute the number of packets in each step:

- *Control messages (N1):* The number of packets in each period p equals the number of nodes in the network, because each node has to send the control packet every period.

$$N1 = N \tag{4.2}$$

- *Multicast client request (N2):* The problem of finding the minimum number of forwarding nodes is NP complete and many approximation approaches have been used [53]. In Bcast, a random rebroadcasting delay is set. In the worst case, a node does not cover all its 2 hops neighbors. Thus, every node has to retransmit the client request.

$$N2 = N \tag{4.3}$$

- *Unicast server reply (N3):* The number of packets exchanged during the unicast server reply is equal to the average number of hops between the service requester and the service provider. The average distance between a service requester and a service provider is computed in [38] as $0.62D$ where D is the diameter of the network. As a result, assuming the transmission ranges of nodes are similar and are equal to t_r. The average number of hops between the provider and the requester is $0.62D/t_r$.

$$N3 = 0.62D/t_r. \tag{4.4}$$

The total number (Ntotal) of packets for SLP in the vehicular network during the time t is:

$$Ntotal = N1 \times \frac{t}{p} + Nk \times (N2 + N3) \tag{4.5}$$

$$= kN^2 + (\frac{0.62Dk}{t_r} + \frac{t}{p})N.$$

Thus $Ntotal$ is $O(N^2)$.

4.5.2.3 Cost for VSDP

In the following, we compute the cost incurred by our proposed VSDP:

- *Cost for service discovery query:* The worst case occurs when the service requested does not exist in the cache of the correspondent Roadside Router. The interval to broadcast a service request follows a random distribution. We suppose $E(x_b)$ the expected interval to send a service request. Thus, the maximum number of service requests $EnReqs$ in one unit of time is:

$$EnReqs = \frac{1}{E(x_b)} \tag{4.6}$$

  The cost for one service query of VSDP (N_query) includes the cost for a propagation of a request message (N_{req}) and the cost for the propagation of a reply message (N_{rep}).

$$N_{req} = (1 - \alpha) \times nRR \tag{4.7}$$

  In the worst case:

$$N_{rep} = nRR \tag{4.8}$$

  where α is the percentage of RRs inside the advertisement zone of a service provider. Thus the total cost for requests during the computation time t is:

$$N\_requests = t/E(x_b) \times (N_{req} + N_{rep}) = (2 - \alpha) \times nRR \times t/E(x_b). \tag{4.9}$$

- *Cost for advertisement messages:* Suppose λ the interval for sending periodic advertisement messages. Each service provider has to send advertisement messages, thus during t,

$$N\_adv = \alpha \times nRR \times \frac{t}{\lambda} \times nSP \tag{4.10}$$

- *Total cost:* The total cost is

$$N\_Total = N\_requests + N\_adv = C_1 \times nRR \tag{4.11}$$

where

$$C_1 = \frac{(2-\alpha) \times t}{E(x_b)} + \frac{nSP \times \alpha \times t}{\lambda} \tag{4.12}$$

Thus, the message complexity of VSDP is $O(nRR)$. As we mentioned earlier, in the best case $nRR = log(N)$, thus the message complexity of VSDP is $O(log(N))$. In the other cases, $nRR = N^c$ where $0 < c < 1$, thus the message complexity of VSDP is $O(N^c)$ as opposed to $O(N^2)$ in SLP.

4.5.3 Service Discovery Cost Functions

To help estimate the cost of VSDP, let us assume λ_{Delay} the one hop delay. As we stated in the system model, we assume a rectangular topology of surface $a \times b$, where a and b are the length and the width of the VANet respectively. We consider that a number of nRR Roadside Routers are uniformly distributed along the vehicular network. Two Roadside Routers are connected directly to each other if the distance d between them is less than or equal to the Roadside Router maximum transmission range tr_0, i.e ($d \leq tr_0$). Thus, we can compute the expected number of hops between any two Roadside Routers. In stochastic geometry, the probability density function (pdf) of a random variable x that denotes a straight line distance between two Roadside Routers, is provided in [12] as:

$$P_X(x) = \frac{4x}{a^2b^2}(\frac{\Pi}{2}ab - ax - bx + 0.5x^2) \tag{4.13}$$

Thus, the expected number of hops $E[H]$ between two Roadside Routers in the network is equivalent to dividing the expected distance $E[x]$ between two RRs by tr_0. The value of $E[x]$ represents a lower bound. For $0 \leq x < b < a$, the $E[H]$ as computed in [12] is:

$$E[H] = 0.521a/tr_0 \tag{4.14}$$

T_{VSDP} and N_{VSDP} combine the cost of one controlled propagation of service request messages and one propagation of service reply messages.

$$T_{VSDP} = E[H] \times delay\_per\_hop \times 2 \Rightarrow \tag{4.15}$$

$$T_{VSDP} = 0.521a/tr_0 \times \lambda \times 2 = 1.042\lambda a/tr_0 \tag{4.16}$$

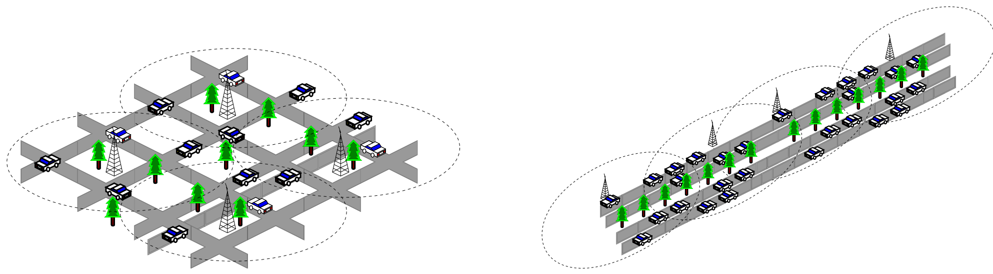
$$N_{VSDP} = number\_of\_controlled\_request\_packets + number\_of\_reply\_packets \Rightarrow \tag{4.17}$$

$$N_{VSDP} = number\_of\_roadside\_routers + E[H] \Rightarrow \tag{4.18}$$

$$N_{VSDP} = nRR + 0.521a/tr_0 \tag{4.19}$$

4.6 VSDP Variants, SLP on VANets: A Comparison

In this section, we compare our scheme to different service discovery protocols through different road-traffic patterns (city and highway patterns). We choose to use SLP on VANets discovery protocol because we want to prove that our VSDP protocol outperforms existing service discovery protocols. We provide an evaluation based on simulation experiments of the multi-channels (MC) hybrid VSDP, the single channel (SC) hybrid VSDP, the multi-channels



(a) Manhattan Mobility Model (b) Highway Mobility Model

Figure 4.7: Different mobility models

(MC) Reactive VSDP, the multi-channels (MC) Proactive VSDP, and SLP on VANets. Before we proceed further, let us describe the protocols used for comparison.

- *SC hybrid VSDP:* consists of the execution of the hybrid VSDP using a unique channel.
- *MC Proactive VSDP:* consists of the execution of VSDP where the advertisement zone's size of every service provider is set to the diameter of the network and where the adaptation process is set off. Thus, in this protocol, every service provider advertises itself in the entire network and every vehicle knows about all the services that exist in the vehicular network.
- *MC Reactive VSDP:* consists of the execution of VSDP where the advertisement zone's size of every service provider is set to zero and where the adaptation process is set off. Thus, in this protocol, every service requester floods the entire network in order to find the appropriate service provider.
- *SLP on vehicular networks:* The description of SLP on vehicular networks has been provided in Section 4.5.2.2.

We recall that Proactive and Reactive VSDP protocols are used in order to prove that our hybrid adaptive technique succeeds in finding the best combination of proactive and reactive approaches through its efficient adaptation process. Through the comparison to the SC hybrid VSDP, we want to show the impact and the importance of channel diversity on the performance of VSDP.

Table 4.3: Common Simulation environment parameters for the city and highway scenarios

Parameter name	**Parameter value**
Wireless medium	IEEE 802.11
Data transmission rate	11Mbps
Transmission range (meters)	200
Service advertisement's life time (seconds)	50
Simulation time(seconds)	1200
Advertisement period (second)	50
Adaptation period (second)	150
Clients vehicles' number	40
Service vehicles' number	10

We choose to evaluate the VSDP's performance using mobility traces that correspond to two alternative road-traffic patterns: *city* and *highway* traffic patterns. The former consists of vehicles moving along a Manhattan mobility model, as shown in Figure 4.7(a), and the latter is comprised of vehicles moving along the highway, as shown in Figure 4.7(b). It is worthy to note that the density in terms of cars, the velocity of cars, and the deployment of RRs are different in every scenario. Both scenarios involve our MC hybrid VSDP, SC hybrid VSDP, MC Proactive VSDP, MC Reactive VSDP, and SLP on VANets as service discovery protocols. In order to obtain these mobility traces, we used two different traffic generators: a Manhattan traffic generator to model the behavior of vehicles in a city [35], and a highway traffic generator to model the behavior of vehicles on the highway [57]. We used NS2 [4] to simulate the compared protocols with the different traffic models. We use the common simulation environment parameters illustrated in table 4.3 for both mobility patterns.

In the course of our experiments, we define the request number as the number of service queries sent by service clients vehicles. Service requests are generated exponentially during the simulation time. We choose to evaluate the performance of our algorithm using the following metrics:

- *Success rate:* indicates the average fraction of successful VSDP transactions over the total number of service queries;

- *Response time:* indicates the average time of successful VSDP transactions. It measures the elapsed time for getting a valid service reply in response to a service request sent by a vehicle. This metric takes into account several factors such as the transmission and message processing delays, just to mention a few;

- *Total bandwidth usage:* computes the total bandwidth consumption by the entire discovery process in bits needed to satisfy the driver's service requests.

4.6.1 Urban Traffic Scenario

In order to evaluate the performance of VSDP in an urban scenario, we used the Manhattan model to emulate the mobility of vehicles on streets gathered on maps [35]. A map contains a number of horizontal and vertical streets. Vertical streets have two different lanes, one directed toward the North and the other toward the South. Meanwhile, horizontal streets have two different lanes for vehicle circulating toward the East or West. Vehicles on the map can move along the vertical or the horizontal streets. At an intersection, a vehicle can turn to the right, turn to the left or keep going straight. A vehicle has a probability of 0.25 to turn to the left, 0.25 to turn to the right and 0.5 to keep going straight. The velocity of a vehicle at the current period of time depends from its velocity at the previous period of time. Thus, the Manhattan mobility model has a high temporal dependence. Moreover, the velocity of a vehicle in one lane is restricted by the velocity of the vehicle preceding it on the same lane. Consequently, the Manhattan mobility model also has a spatial dependence. The Manhattan mobility model imposes geographic restrictions on the mobility of vehicles on the streets, but it also gives a vehicle freedom to change its direction.

Simulation Experiments

In this section, we evaluate and compare the performance of our algorithm and assess the success rate and the response time obtained by our algorithm. Then, we discuss the message overhead of our VSDP schemes and compare it to the theoretical results. We use the simulation parameters illustrated in table 4.4, specific to the city pattern. We varied the number of

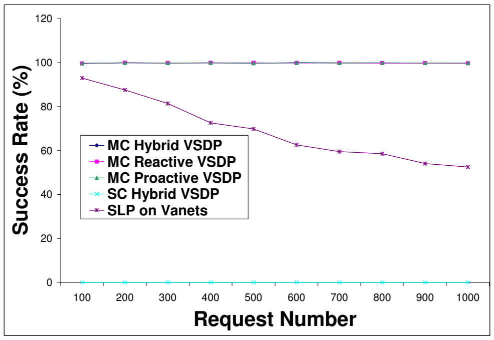
Table 4.4: Simulation parameters for the city scenario

Parameter name	Parameter value
Vehicle's Speed (meter/second)	[0..20]
Simulation area ($meter^2$)	1,250,000
Average vehicles density (vehicles/Km^2)	334
Roadside routers number	25

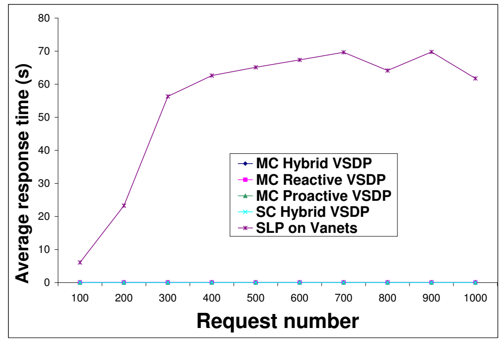
vehicle requests between 100 and 1000. The following experimental data were obtained by averaging 15 runs with a confidence of 95 percent. The results are summarized in Figure 4.8.

Let us now turn to our results. The success rate for the city traffic scenario is shown in Figure 4.8(a). As we can depict, the success rates of the MC hybrid VSDP, the MC proactive VSDP, and the MC reactive VSDP are more than 98 percent respectively for a number of service requests ranging from 100 to 1000. The main cause of unsuccessful service queries is due to the dropping of service requests when vehicles cannot connect to the roadside backbone. Our MC hybrid VSDP outperforms SLP on VANets. In fact, as we can depict from Figure 4.8(a), the success rate of SLP on VANets decreases with the increase of the number of service requests and is less than 60 percent for 1000 service requests. The success rate of the SC hybrid VSDP is very low and even null. This is mainly due to our design concept of VSDP protocols where request and reply messages are sent and received at the same time through different interfaces using different channels. Consequently, if a single channel is set on the two different interfaces, and if request and reply messages are sent simultaneously through the different interfaces, then packets are prone to collide and a service reply would never be sent back to the service requester.

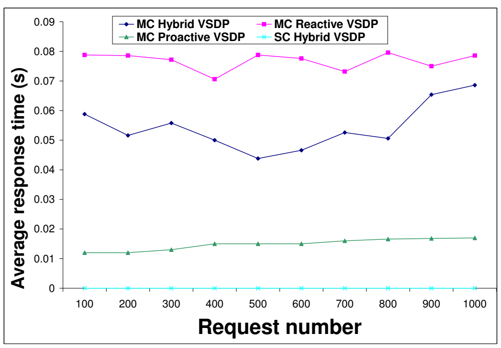
Figure 4.8(b) portrays the average response time of the MC hybrid VSDP, the SC hybrid VSDP, the MC proactive VSDP, the MC reactive VSDP, and SLP on VANets. We depict that the average latency of the different approaches of VSDP outperform considerably the average latency of SLP on VANets with an average of 10 milliseconds for hybrid VSDP and up to 60 seconds for SLP when the number of requests is 1000. This also meets our expectations because we took in consideration the channel diversity in our discovery process. In fact, in our VSDP protocol, we use the multi-channels and multi-interfaces feature. This permits to



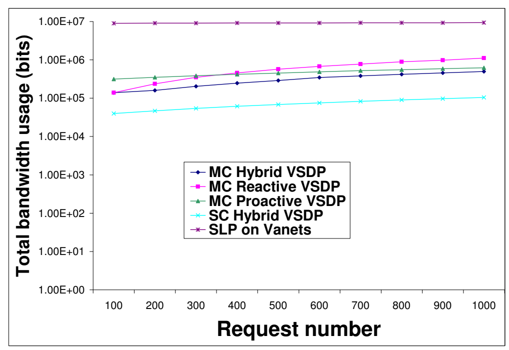
(a) Success Rate comparison



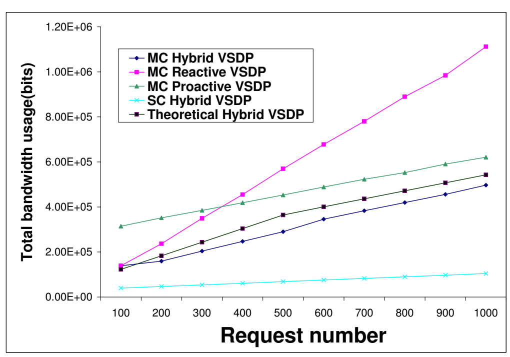
(b) Average response time comparison



(c) Average response time comparison of VSDP variants



(d) Total bandwidth usage comparison



(e) Total bandwidth usage comparison of VSDP variants

Figure 4.8: Performance evaluation and comparison of VSDP in a city traffic model

decrease considerably the response time because messages can be sent and received at the same time on different channels and the traffic congestion is decreased on single channels. However, the adapted SLP for VANets does not use channel diversity. Thus, the response time is much higher as it is shown in the curve in Figure 4.8(b). Another factor that contributes to the low response time of our VSDP is the network-layer based design of our protocol. In fact, as we described earlier, our MC VSDPs implement a lightweight advertisement and discovery technique that aims to save the network resources by finding simultaneously a service provider as well as its route.

In order to be able to evaluate the average response time of the different approaches (MC reactive, MC proactive, MC hybrid and SC hybrid) of VSDP, we plot the graphs on Figure 4.8(c). These graphs show that the proactive approach has the lowest average response time of 15 milliseconds for 1000 requests, and that the reactive approach has the highest average latency of 80 milliseconds for 1000 requests. The hybrid approach has an average latency where the values are between the proactive and the reactive approaches values for a number of requests ranging from 100 to 1000. Thus, we can conclude that our adaptation mechanism finds a good tradeoff between the reactive and the proactive approaches.

Figure 4.8(d) portrays the bandwidth usage of the MC hybrid VSDP, the SC hybrid VSDP, the MC Reactive VSDP, the MC Proactive VSDP, and the SLP on VANets. SLP on VANets has the highest bandwidth usage that ranges from 9,037,232 bits for 100 requests to 9,357,859 bits for 1000 requests. This high bandwidth usage shows the non scalability of SLP on VANets. On the other hand, we depict from Figure 4.8(d) that VSDPs are scalable compared to the SLP on VANets. Figure 4.8(e) illustrates the bandwidth usage curves of the MC and SC VSDPs, as well as the curve resulting from the theoretical computation of the bandwidth usage of VSDP. We notice that the MC hybrid VSDP has the lowest bandwidth usage compared to the MC reactive VSDP and the MC proactive VSDP when the number of requests ranges from 100 to 1000. This proves again, that our adaptation mechanism was successful and efficient in finding the tradeoff between the reactive and the proactive approaches while determining the size of service providers' advertisement zones. The bandwidth usage of the MC Proactive VSDP is higher than that of the MC reactive and hybrid VSDPs, because in the proactive approach, SPs advertise themselves in the whole vehicular network every advertisement period. The bandwidth usage of the SC VSDP is low compared to the MC discovery approaches because discovery queries with the SC VSDP are not successful and are dropped at an early stage of their processing. The graphs in Figure 4.8(e) show that the theoretical and the simulation based MC hybrid VSDPs bandwidth usage curves are very close to each other and have the same behavior as we were expecting.

4.6.2 Highway Traffic Scenario

In order to evaluate the performance of VSDP in a highway, we use the traffic generator described in [57] and the code provided in [56] to generate the tcl files fed to NS2. We generate 3 km long bidirectional highways with four straight lanes. Vehicle speed varies between 60 and 100 km/h. The speed of a vehicle on the highway depends on that of the preceding vehicle. The traffic generator described in [57] provides the rules of movement for cars along the highway when they face obstacles or slower cars. The simulation time is set to 1200 seconds. In this scenario, we choose the 802.11 as wireless medium with a data transmission rate of 11Mbps and a transmission range of 200 meters. When a vehicle enters in the highway, it chooses a random time to initiate its request.

Simulation Experiments

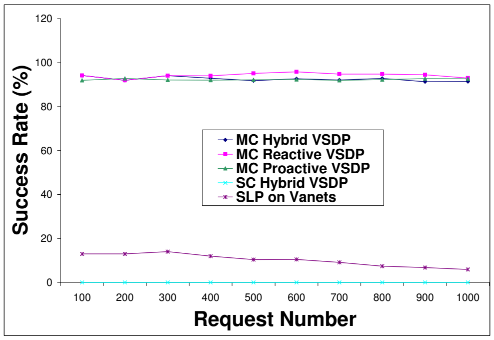
In this section, we evaluate the performance of our algorithm and assess the success rate and the response time obtained by our algorithm. Then, we discuss the bandwidth usage of our VSDP schemes. We use the simulation parameters illustrated in Table 4.5, specific to the highway pattern. We vary the number of vehicle requests between 100 and 1000. The following experimental data were obtained by averaging 15 runs with a confidence of 95

Table 4.5: Simulation parameters for the Highway scenario

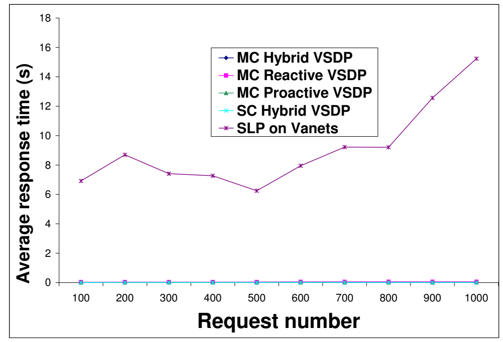
Parameter name	Parameter value
Vehicle's Speed (km/hour)	[60..100]
Simulation area (*meter*^{2})	300,000
Average vehicles density (vehicles/Km^2)	334
Roadside routers number	10

percent. The results are summarized in the graphs in Figure 4.9. Let us now turn to our results. Figure 4.9(a) portrays the success rate of the MC hybrid VSDP, the SC hybrid VSDP, the MC proactive VSDP, the MC reactive VSDP, and SLP on VANets. As we can depict, the success rate of the MC hybrid VSDP, the MC proactive VSDP, and the MC reactive VSDP are more than 90 percent. Our MC hybrid VSDP outperforms SLP on VANets. In fact, as we deduce from Figure 4.9(a), the success rate of SLP on VANets is less than 20 percent and decreases with the increase of the number of service requests to less than 10 percent for 1000 request. Moreover, we notice that the success rate of SC hybrid VSDP is very low. It is less than 10 percent. This is mainly due to our implementation design. In the design of our VSDP protocols, messages are sent and received simultaneously on different interfaces through different channels. Since SC hybrid VSDP uses only one channel, packet collisions are increased and are unavoidable for the SC hybrid VSDP. Thus, request messages and reply messages are very prone to collide and to be discarded, which explains the low success rate in the SC hybrid VSDP.

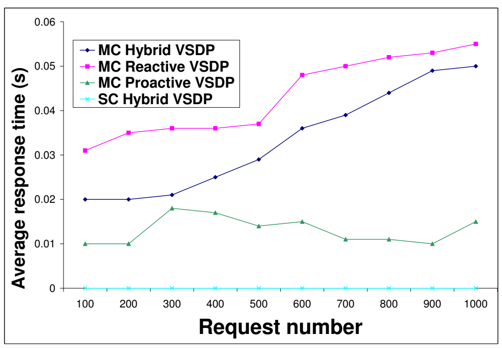
Figure 4.9(b) describes the average response time of the MC hybrid VSDP, the SC hybrid VSDP, the MC proactive VSDP, the MC reactive VSDP, and SLP on VANets. We depict that the average response time of the different approaches of VSDP outperform considerably the average response time of SLP on VANets. In fact, the average response time of MC hybrid VSDP has a value of 20 milliseconds for 100 requests increasing to 50 milliseconds for 1000 requests. However, for the SLP on VANets the average response time ranges from 6 seconds for 100 requests to 16 seconds for 1000 requests. This meets our expectations because we took on consideration the channel diversity in our discovery process. As we explained for the city traffic scenario, in our VSDPs, we use the multi-channel and multi-interfaces feature.



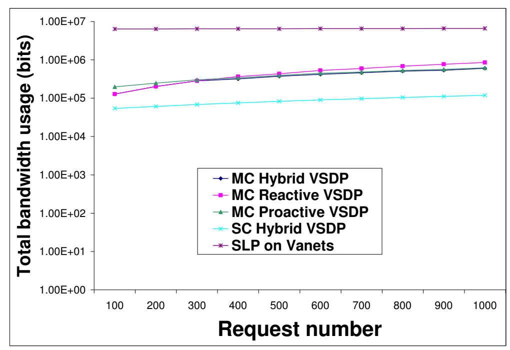
(a) Success Rate comparison



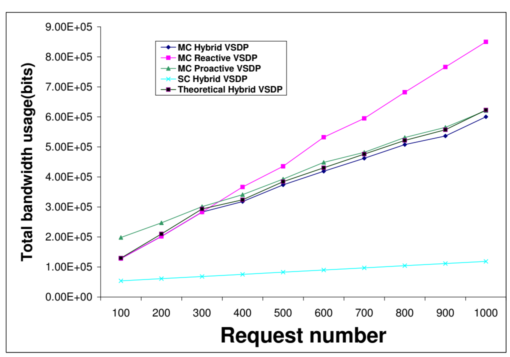
(b) Average response time comparison



(c) Average response time comparison of VSDP variants



(d) Total bandwidth usage comparison



(e) Total bandwidth usage comparison of VSDP variants

Figure 4.9: Performance evaluation and comparison of VSDP in a Highway traffic model

This permits to decrease considerably the response time because messages can be sent and received at the same time on different channels and the traffic congestion is reduced on single channel. We notice that SLP on VANets has lower response time in the highway scenario than in the city scenario. This is mainly related to the success rate. In the city scenario, the success rate of SLP is higher that that for the highway scenario. Thus, the average response time is computed for a higher number of successful requests. The SLP on VANets does not use channel diversity and thus the response time is much higher as it is shown in the curve in Figure 4.9(b). Another factor that contributes to the low response time of our MC VSDPs is the network-layer based design of our protocols. In fact, as we described earlier, our VSDPs implement a lightweight advertisement and discovery technique that aims to save the network resources by finding simultaneously a service provider as well as its routing information. In order to be able to evaluate the average latency of the different approaches (reactive, proactive and hybrid) of VSDP, we plot the graphs in Figure 4.9(c). All these approaches have an average response time less than 60 milliseconds. Moreover, we notice a variation in the values of the average response time for the different approaches.

The graphs in Figure 4.9(c) show that the MC proactive approach has the lowest values, the MC reactive approach has the highest values, and the MC hybrid approach values are between the reactive and the proactive approaches when the number of requests ranges from 100 to 1000 requests. This behavior was expected because in the proactive approach, service requesters know about the existing service providers, thus they can find them quickly. For the reactive approach, service providers do not advertise themselves in the VANet, consequently, it takes longer to discover them. Finally, in the hybrid approach service providers advertise themselves in an adaptively determined zone, as a result, it takes less time than the reactive approach and more time than the proactive approach to find the requested service. As a consequence, we can conclude that our proposed adaptation mechanism finds a good tradeoff between the reactive and the proactive approaches. For the SC hybrid approach, the average response time is not meaningful because its correspondent success rate was very low.

Figure 4.9(d) portrays the bandwidth usage of the MC hybrid VSDP, the SC hybrid VSDP, the MC reactive VSDP, the MC proactive VSDP, and SLP on VANets. As we can depict, the MC hybrid VSDP outperforms the SLP on VANets. The bandwidth usage of SLP on VANets increases drastically which can limit the scalability of SLP. However, the bandwidth usage of our MC hybrid VSDP increases reasonably with the increase of the number of service requests. In addition, we can depict from Figure 4.9(e) that the bandwidth usage of the MC hybrid VSDP is lower than the bandwidth usage of the MC proactive VSDP and that of the MC reactive VSDP. This proves the good performance of our protocol and the efficiency of our proposed adaptation mechanism that succeeds to find the best tradeoff between the proactive and the reactive approach to permit the scalability of the hybrid VSDP while guaranteeing high success rate and low response time. In addition to the curves of VSDP approaches, Figure 4.8(e) illustrates the theoretical curve of the hybrid VSDP. It shows that our theoretical and simulation based curves have the same behavior and almost the same values correspondent to the bandwidth usage. More precisely, the theoretical values are slightly higher than the simulation based values because in this latter, the adaptation mechanism is processed during all the duration of the simulation. However, for the theoretical curve, we choose to plot the value correspondent to the overall best combination of the reactive and the proactive zones.

4.7 Summary

In this chapter, we proposed a scalable hybrid adaptive service discovery protocol for vehicular networks (VSDP). We showed that our VSDP is well suited for congested highly mobile vehicular networks or intermittent VANets. A comparative study of performance of VSDP and other service discovery approaches and protocols with different road traffic patterns has shown that all the three VSDP approaches have outperformed considerably the SLP on VANets in terms of high success rate and low response time of service queries. With respect to the bandwidth usage, hybrid VSDP succeeded through our proposed adaptation mechanism to find the best tradeoff between the reactive and the proactive components. Moreover, hybrid VSDP has demonstrated a considerable lower bandwidth usage than the SLP on VANets, which permits the scalability of VSDP. Finally, we would like to state that our adaptation mechanism could be applied to existing hybrid service discovery protocols. The main weakness of protocols in hybrid approach is how to determine an efficient size of the service provider advertisement zone. Thus, instead of using a fixed size, we believe that the use of our efficient adaptation mechanism could improve the performance of existing protocols.

Chapter 5

Location-aware Service Discovery Protocols: LocVSDPs

5.1 Introduction

Several open research challenges are delaying the efficient and widespread deployment and management of many applications, such as security and safety; and traffic information and service location applications, on VANets. One of these challenges comprises how drivers or passengers, which are well-known for their large-scale and high-mobility nature, could discover service providers located in desired regions in the VANet. In the following, we discuss our contribution and research challenges for this chapter. Then, we present our contribution and objectives. Finally, we show the chapter organization.

5.1.1 Motivation and Research Challenges

In mobile ad hoc networks and VANets, few existing projects have focused on proposing location-based service discovery protocols [81, 74, 44, 69, 26, 36, 41, 21, 5, 25, 45]. Dikaiakos *et al.* in [25] and Klimin *et al.* in [45] proposed distributed location-based service discovery protocols with the characteristics of VANets in mind. Their protocols were designed for high speed type of networks as opposed to context-aware protocols cited in [81, 74, 44]. Moreover, the protocols in [25, 45] do not rely on service code migration as opposed to the protocols cited in [69, 26, 36], and they are distributed and self-configurable as opposed to protocols cited in [41, 21, 5]. However, the main drawback of these protocols [25, 45] is their limited scalability in highly dense VANets with a large number of service requests due to their infrastructure-less architecture.

5.1.2 Contribution and Objectives

In this chapter, we propose efficient and scalable infrastructure-based context-aware and location-based service discovery protocols for VANets: *(i)* The election-based LocVSDP *(EB-LocVSDP)*; *(ii)* and its variant the *Naive LocVSDP*. Then we enhance the EB-LocVSDP with fault tolerance techniques, and with load balancing and QoS based mechanisms. The protocols infrastructure relies on clusters of wireless Roadside Routers distributed in the VANet. The distribution of clusters depends on the applications requirements in the VANet. Clusters of Roadside Routers are mainly formed around and nearby service providers, to help the management of service queries. Our proposed protocols provide scalable framework for the discovery of time-sensitive and location-based services in VANets. They help drivers and passengers to find services located in a desired region, that we will call *Region of Interest (RI)* from now on, such as restaurants with their menus or gas stations with their price lists. For this purpose, the driver or passenger specifies the region of interest where he wants to find a service. The main advantages that help the scalability of our LocVSDP are: *(i)* they rely on a cluster-based infrastructure, *(ii)* they are integrated into the network layer, *(iii)* they use channel diversity, and *(iv)* they find efficiently services located in the area specified in a driver's or passenger's request.

Our proposed protocols rely on a cluster-based infrastructure where clusters could be formed *(i)* nearby service providers, *(ii)* in the congested areas, or *(iii)* along area with intermittent connections to help the management of service queries and to guarantee the network connectivity. LocVSDPs find the service provider and its routing information simultaneously which results in overall bandwidth savings. They use diverse channels to exchange discovery and routing packets for an efficient usage of the radio spectrum, an increase of the wireless network capacity, and a delay reduction in service queries. Our proposed location-based service discovery protocols find services located in the region of interest specified in the driver's or passenger's request using efficient location-based propagation of the request and efficient computation of the reply.

5.1.3 Chapter Organization

The remainder of this chapter is organized as follows. Section 5.2 provides our problem statement and system model. Section 5.3 contains a detailed description of our proposed location-based service discovery protocols (EB-LocVSDP and Naive LocVSDP). Section 5.4 presents the complexity computation model and the cost function of our proposed EB-LocVSDP scheme. Section 5.5 presents the performance evaluation study of our protocols and their comparison to a related work. Section 5.6, presents and evaluates fault tolerance mechanisms for the EB-LocVSDP. Section 5.7 presents and evaluates load balancing and QoS based techniques for the EB-LocVSDP. Finally, Section 5.8 concludes the chapter.

5.2 Problem Statement and System Model

In the following, we formulate the location-based service discovery problem, then we discuss our system model.

5.2.1 Problem Statement

A service discovery mechanism is needed to allow drivers and passengers requesting services in a predetermined geographic area, and vehicles or roadside components providing services to discover each other in a VANet. In the rest of the chapter we will refer to the predetermined geographic area as the "Region of Interest (RI)". A driver or passenger may need a particular service that may be provided by one or more road components located inside the RI. The driver or passenger needs a mechanism to find out which vehicular component provides the desired requested service.

The service discovery mechanism should perform without any prior configuration. Moreover, the discovery mechanism needs to work in a large-scale VANet, where the number of location-based service requests is high. An example of location-based services, where the geographic location of services is specified in the service request message, could be: free parking spots, restaurants, gas stations, etc.

5.2.2 System Model

In our system model, we consider two basic elements: Roadside Routers (RRs) and Road Vehicles (RVs).

1. ***Roadside Routers (RRs)*** are fixed or have low mobility, and they have routing capabilities. They are clustered into one or more RRs. RRs in one cluster are connected to each other and form an IEEE 802.11-based wireless Roadside Cluster (RC_i) in a cluster i. We assume in our system that each RR is equipped with two radio interfaces and uses diverse channels. In fact, the capacities and capabilities of RRs are enhanced if diverse channels are used efficiently between RRs. For packet routing between RRs, we implement an enhanced version of the CLA-S routing protocol [40]. In this enhanced version, we consider the usage of multiple interfaces and multiple channels.

2. ***Road Vehicles (RVs)*** are the vehicles moving along the roads. They are characterized by their high mobility and their increased density. They have in general devices integrated into them and may carry zero or one wireless interface. Vehicles without wireless interface will not use our service discovery protocol, thus they will not affect our mechanism. Vehicles equipped with the wireless interface, use the latter for the ad hoc communica-

tion between each other and with the RRs. The IEEE 802.11 protocol is used for the wireless communication between vehicles and between vehicles and RRs.

The wireless connection between two vehicles could be performed through three different ways: *(i)* directly, *(ii)* through intermediate vehicles, or *(iii)* through RRs in a cluster. Vehicles can reach the wireless clustered RRs in multi-hops using intermediate vehicles. VANets have particular mobility models different from MANets, where node mobility is not completely random, but restricted by roads. Wireless routing between vehicles, RRs, and between vehicles and RRs is based on an enhanced version of the Connection less Approach on Street (CLA-S) [40] routing protocol. CLA-S is a packet based routing protocol dedicated to VANets with high mobility of vehicles. It uses the geographic location information of destination vehicles for a hop by hop routing till the destination.

In our system model, vehicles move on a two dimensional plane. We use a rectangular topology where the area is $a \times b$ (a and b are the length and the width of the VANet, respectively, and b could be equal to a). RRs are distributed randomly along the VANet around *(i)* the known service providers, *(ii)* the predictable congested areas, *(iii)* and the predetermined intermittent areas. The random distribution of RRs around the known service providers permits to prevent bottlenecks when many service requests are directed to the same region of interest. We distribute RRs around known service providers. However, our protocol can be processed even if new service providers are introduced into the VANet. RRs are distributed around predictable congested area in order to reduce the dropping of service requests when the network is congested and there are many service requests. This helps the scalability of our proposed discovery scheme. Moreover, RRs are randomly distributed around intermittent areas in the VANet to permit the connectivity in the VANet. RRs in the communication range of each others form a cluster. The total number of RRs used in our system is *nRR*.

5.3 Description of the Proposed LocVSDP Algorithms

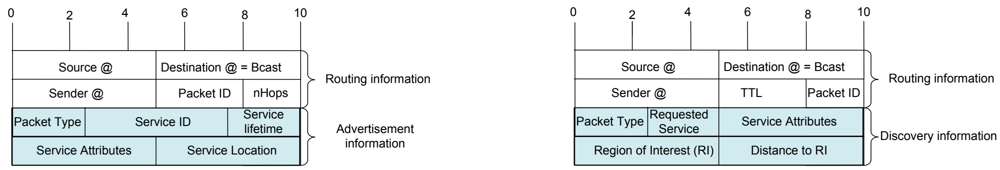
In this section, we describe our proposed protocol: the Election-based Location-based Vehicular Service Discovery Protocol (EB-LocVSDP), then we describe its variant (The Naive-LocVSDP). We use the pseudocode illustrated in Algorithms 5.3.1, 5.3.2, and 5.3.3 for the formal description of our EB-LocVSDP. Our protocol is specially designed to find services located in a predetermined region specified by the driver or passenger: *the region of interest (RI)*. The main purpose of our protocol is to help drivers and passengers to locate services close to the destination of the driver or in a predetermined region. Our proposed protocol is infrastructure-based. Roadside Routers (RRs) are randomly distributed in the VANet and they are clustered. RRs in the communication range of each other form a cluster. They are fixed or have low mobility, and they have routing capabilities. RRs are equipped with multiple radios and use diverse channels. RRs capacities and capabilities are improved when an efficient utilization of channels is taken into consideration. Road Vehicles (RVs) are the cars circulating along the roads. They are characterized by their high mobility and their increased density. EB-LocVSDP can be divided into four separate phases: *(i) service advertisement*, *(ii) service request propagation*, *(iii) leader election and service reply generation*, and *(iv)service reply propagation*. Before we proceed further, let us discuss the features and characteristics of the proposed LocVSDPs.

5.3.1 LocVSDPs Features and Characteristics

In the following, we present and discuss the main features and characteristics of the proposed LocVSDPs.

(a) **Context Aware and Location Based Discovery:** Our proposed protocols are context aware and location based. The service requester specifies the location of the desired service. Consequently, not all the services in the VANet that satisfy the driver's or passenger's request are returned to the requester but only service providers that satisfy the request and that are located in the desired region of interest specified by the requester. This permits to save the bandwidth usage in the network and to improve its capacity. In fact, unnecessary and irrelevant service replies from providers outside the region of interest will not consume and waste the network bandwidth.

(b) **Cluster-Based Infrastructure Support:** Our LocVSDPs rely on a cluster-based infrastructure for the discovery of location based services. We believe that the infrastructure support is suitable for large-scale VANets. Moreover, infrastructure that relies on clusters is very convenient for location based types of applications, where clusters are formed around either the most congested sites in the network or toward some sites in the network where wireless communications cannot be established easily because of the intermittent connections. A cluster is formed by RRs in the communication range of each others. A RR that is not in the communication range of any other RR can form by itself a cluster. Thus, in our design, a cluster can contain one or more RRs. The number and the distribution of clusters in the VANet depend on the network applications requirement. For example, clusters are mainly deployed into the locations that contain many service providers. This permits an efficient management of service queries and increases the bandwidth usage in the network. Inside a cluster, RRs could be distributed uniformly or randomly. Wireless inter-clusters and intra-clusters communications are based on the CLA-S [40] routing protocol. The communication between clusters could be done through vehicles. As mentioned earlier, clusters are mainly formed around and toward service providers. As a result, in our LocVSDPs, the discovery messages are mainly exchanged through RRs belonging to the cluster-based infrastructure. This reduces considerably the network traffic when compared to infrastructure-less discovery protocols. Moreover, a highly mobile VANet with infrastructure support permits a level of stability during the communication between service providers and service requesters. However, service discovery queries in highly mobile infrastructure-less VANets could be dropped so often due to the high mobility of vehicles and the occurrence of partitions.



(a) L-VPD packet format

(b) L-VRD packet format

Figure 5.1: Formats of L-VPD and L-VRD packets

(c) **Network Layer Support:** Our proposed LocVSDPs are network layer based. By this we mean that service information messages are piggybacked into the routing messages, which permits a lightweight discovery of service providers. In fact, in large-scale congested VANets, the limited network resources could be drastically affected due to the redundant flooding of service discovery messages and routing messages. If we piggyback service discovery messages into the routing messages, then a service provider as well as its routing information are found. This saves considerably the network resources and permits the scalability of the LocVSDPs. As a routing protocol, we use a modified version of the Connectionless Approach for Streets CLA-S [40] routing protocol. This latter is packet based and handles links break issues. In fact, in this protocol communication packets are sent by RRs or vehicles inside a predetermined forwarding area. For this purpose, location information for the determination of the forwarding zone are integrated into the routing packets. Only road components (RRs or vehicles) located inside the forwarding zone can decide whether or not they retransmit the received packets. Moreover, the routing protocol CLA-S [40] provides many routing paths between a service provider and a service requester for data communication. Our modifications to the original CLA-S [40] routing protocol consist of integrating channel diversity and piggybacking service discovery information into the routing messages. In order to handle the channel diversity feature at the RRs, we manage to specify the channels on each radio interface so that transmission and reception of messages at a RR are handled through different channels. The routing packets formats after piggybacking service discovery information are illustrated in Figure 5.1.

(d) **Multi-Channels and Multi-Interfaces LocVSDP:** In our LocVSDPs, the communication between RRs uses multiple radio interfaces and diverse channels for the exchange of discovery and routing packets. The purpose from using channel diversity is mainly to decrease the congestion on single channels and thus decreasing the service discovery transaction delay. In [32], Gupta *et al.* proved that the capacity of wireless ad hoc networks is decreased due to the interference between multiple simultaneous transmissions. Moreover, the capacity of relaying RRs that use a single radio is halved. Whereas, RRs that have multiple interfaces can receive and transmit at the same time. They can exploit the radio spectrum efficiently through the usage of multiple channels at the same time. Consequently, equipping RRs with multiple radios using diverse channels enhances the capacity of wireless networks and promises good results as stated in [7] and [8]. Finally, it is worthy to note that the cost of IEEE 802.11 radios is becoming low and that equipping RRs with multiple radios is feasible and not anymore an issue. In our LocVSDPs, we implement the multi-channels and multi-interfaces feature at the RRs. We assume that every RR is equipped with at least two interfaces (interface1 and interface2). Every interface uses a fixed channel different from the one used in the other interface of the same Roadside Router. In order to guarantee the connectivity in the VANet and to avoid channel switching costs, we assume that interface1 is set to the fixed channel1 and that interface2 is set to the fixed channel2. The usage of the different interfaces is handled at the routing module. This latter determines which interface is used for the propagation of LocVSDPs messages. We manage the usage of interfaces in such a way that sending and receiving messages are performed using different channels.

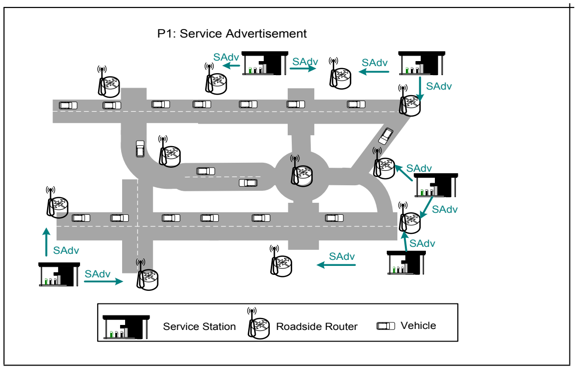
5.3.2 EB-LocVSDP phases

The four phases of our proposed EB-LocVSDP are described below:

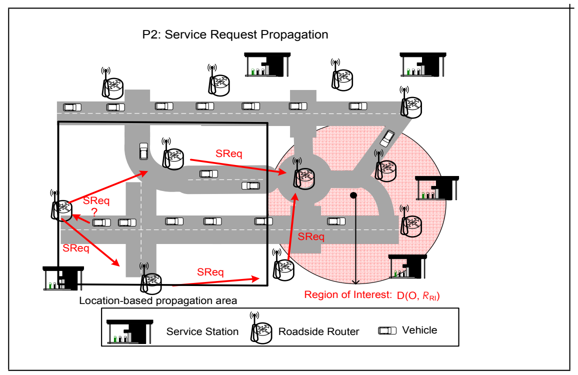
- **Phase 1: Service Advertisement:** We suppose that we have three types of services: (i) fixed services, (ii) moving services, and (iii) Migratory services.

- *Fixed services* have a predetermined location. Their position does not change over the time. Examples of services in this category could be: restaurants with their menus, gas station with their price list, parkings with available spots, etc.
- *Moving services* are the services provided by the vehicles on roads. The location of these services depends on the location of the moving vehicle. Example of services in this category could be: music sharing, game sharing, file sharing, etc.
- *Migratory services* have a fixed location but they are provided by moving vehicles. When vehicles are moving around the fixed location, they provide the service, when they are far from the predetermined location, the service migrate to provider vehicles close to the predetermined location. Examples of services in this category could be sightseeing in a predetermined area, traffic condition monitoring in a specific area, accident or disaster monitoring in a specific area, etc. Our proposed EB-LocVSDP can be used to discover any of the three types of service mentioned above.

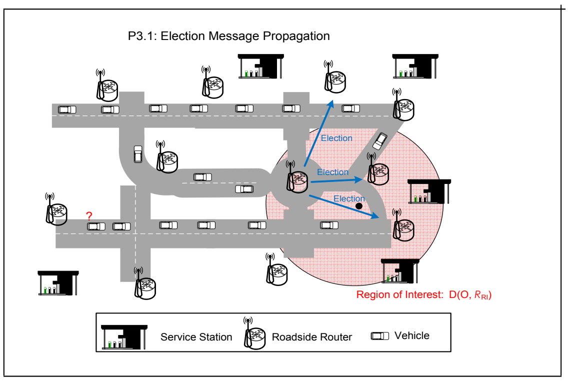
The service advertisement phase is illustrated in Figure 5.2(a). Service providers advertise themselves by sending advertisement messages in their ranges. Advertisement messages are intercepted by neighboring Roadside Routers if they exist. The formal description of the service advertisement phase is elaborated in the Algorithms 5.3.1 and 5.3.2 (Case A, D and E). The format of an advertisement message is illustrated in Figure 5.1(a). An advertisement message or a Location-based Vehicular Proactive Discovery (L-VPD) packet contains both routing information (source address, destination address, sender address, packet ID and time-to-live) and service discovery information (packet type, service ID, service lifetime, service attributes, and service location). An advertisement message or L-VPD packet is intercepted by the Integrated Modules of the neighboring Roadside Routers. In each Roadside Router, the Integrated Module that receives the service advertisement message or L-VPD packet separates the discovery information from the routing information. Discovery information is processed by the



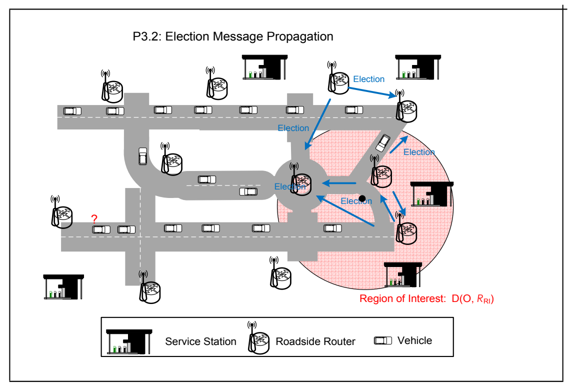
(a) Phase 1



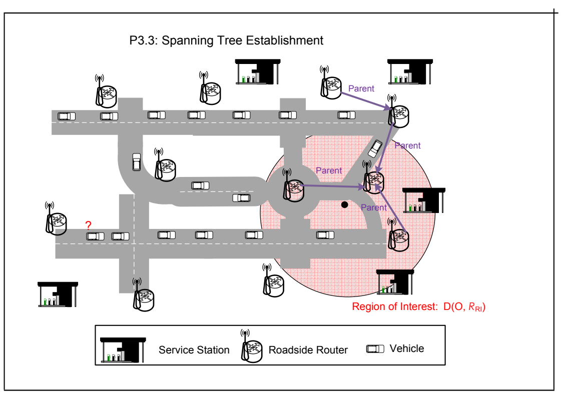
(b) Phase 2



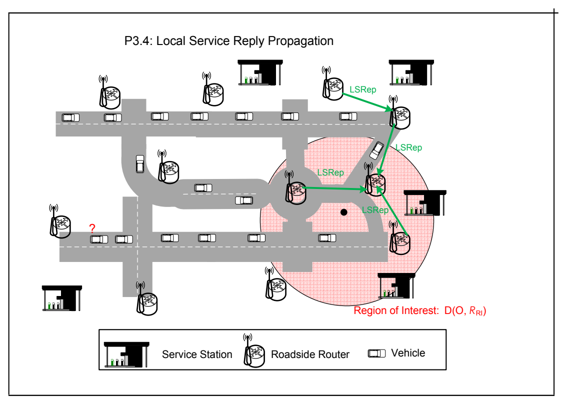
(c) Phase3.1



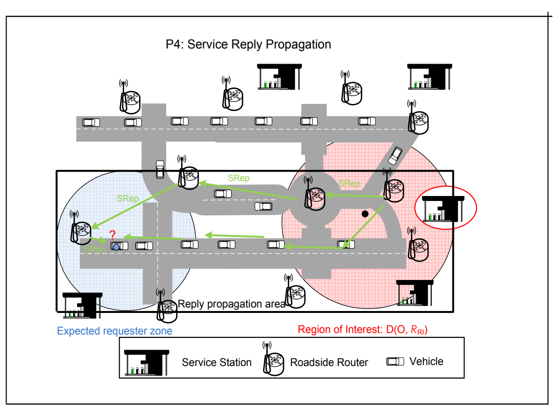
(d) Phase 3.2



(e) Phase 3.3



(f) Phase 3.4



(g) Phase 4

Figure 5.2: EB-LocVSDP Phases

Algorithm 5.3.1 Protocol running at the Vehicle Vi

Case A. V_i has a service to advertise:
Step A1. Periodically (Every $\Delta_{advertisement}$): Broadcast the advertisement message
Case B. V_k needs to find a service s:
while Step B0. N<3 times ***or*** NO service reply $SRep$ message is received **do**
  Step B0.0. V_k sends a location-based service request $L - SReq$ to its neighboring RRs and Vehicles
  Step B0.1. Wait for a predetermined period of time.
end while
if Step B1. ($L - SRep$ message received) **then**
  Step B1.0. GOTO Step C0.
else
  Step B2.0. Exit: Service not found
end if
Case C. V_k receives a service request message $L - VRD$ and does not have a RR in its range:
Step C0 V_k checks whether or not it is inside the requested Region of Interest $RI(C(O_{RI}, R_{RI}))$:
if Step C1 $(xV_k - xO)^2 + (yV_k - yO)^2 \geq R_{RI}^2$ **then**
  Step C1.0 (V_k is outside RI)
  if Step C1.1 $\sqrt{((xV_k - xO)^2 + (yV_k - yO)^2)} < L - VRD-> distance\_to\_RI$ **then**
    Step C1.1.1 V_k generates a Location-based Vehicular Reactive Discovery $L - VRD$ packet.
    Step C1.1.2 V_k Propagates the $L - VRD$ packet to the neighboring RR or Vehicle.
  end if
else if Step C2$(xV_k - xO)^2 + (yV_k - yO)^2 < R_{RI}^2$ **then**
  Step C2.0 V_k is inside the Region of Interest RI:
  if Step C2.1 V_k is the requested service provider **then**
    Step C2.1.0 Extract location information of the requesting vehicle from $L - VRD$ packet
    Step C2.1.1 Send service reply $L - SRep$ message to the requester using CLA-S
  else
    Broadcast $L - VRD$ packet
  end if
end if
Case CC. V_k receives a service reply :
if Step CC0. V_k is the service requester **then**
  Step CC0.0 Store service provider information.
  Step CC0.1 Extract service provider routing information.
  Step CC0.2 Establish the connection with the provider.
else
  Step CC1. Route the service reply message using CLA-S.
end if

Service Module and the routing information is processed by the Routing Module. The Service Module adds the service information to its service table. If the service exists already in the service table, the Service Module updates the information of the existing service. The format of the service table is illustrated in Figure 5.3(b). The field *CandidateRI* in the service table refers to the regions of interest for which *serviceID* has been selected. This information is useful for answering new service requests from previously cached service replies. The Routing Module adds the routing information to its routing table. If the routing entry exists already in the routing table, then the Routing Module updates the entry routing information. The format of the routing table is illustrated in Figure 5.3(a). It is worthy to note that service advertisement messages are processed by RRs only.

Algorithm 5.3.2 PROTOCOL RUNNING AT THE ROUTER RRI

Case D. receive a service advertisement message $SAdv\_msg$ from a vehicle V_i:
Step D0. The Service Module updates the service information table.
Step D1. The Routing Module updates routing table.
if Step D2. ($nhops > 0$) **then**
  Step D2.0. $nhops = nhops - 1$.
  Step D2.1. The Integrated Module generates the appropriate L-VPD message.
  Step D2.2. Forward the L-VPD message.
end if
Case E. receive a L-VPD message from a Roadside Router RR_j:
Step E0. Decode the packet by separating the routing information from service information.
Step E1. The Service Module updates the service information table.
Step E2. The Routing Module updates the routing table.
if Step E3. (nhops > 0) **then**
  Step E3.0. $nhops = nhops - 1$.
  Step E3.1. The Integrated Module forwards the L-VPD message.
end if
Case F. receives a location-based service request message $L - SReq\_msg$ from V_k:
Step F0. The Service Module checks the service description.
if Step F1. service exists in service information table and the service is candidate in the requested RI **then**
  Step F1.0. return a service reply message $SRep$ with routing information from the Routing Module to V_k.
else
  Step F2.0. The Service Module solicits the Routing Module to collaborate in finding the requested service.
  Step F2.1. The Routing Module generates the appropriate location-based route request $L - RReq$ packet.
  Step F2.2. The Integrated Module generates a Location-based Vehicular Reactive Discovery $L - VRD$ packet.
  Step F2.3. Propagate the L-VRD packet to the neighboring Roadside Routers.
end if
Case G. receive a L-VRD message from Roadside Router RR_j:
Step G0. decode the packet by separating the routing information from service information.
Step G1. Send location-based service request L-SReq to Service Module.
if Step G2. service exists in service information table and the service is candidate in the requested RI **then**
  Step G2.0. send a service reply message L-SRep with routing information to RR_j.
else
  Step G3.0. The Routing Module checks whether or not the current Roadside Router is inside the requested Region of Interest $RI(C(O_{RI}, R_{RI}))$:
  if Step G3.1. $(xRR - xO)^2 + (yRR - yO)^2 \geq R_{RI}^2$ **then**
    Step G3.1.0 (RR is outside RI)
    if Step G3.1.1 $\sqrt{((xRR - xO)^2 + (yRR - yO)^2)} < L - VRD- > distance\_to\_RI$ **then**
      Step G3.1.1.0 The Routing Module generates the appropriate location-based route request L-RReq packet.
      Step G3.1.1.1 The Integrated Module generates a Location-based Vehicular Reactive Discovery L-VRD packet.
      Step G3.1.1.2 Propagate the L-VRD packet to the neighboring Roadside Routers.
    end if
  else if Step G3.2 $(xRR - xO)^2 + (yRR - yO)^2 < R_{RI}^2$ **then**
    Step G3.2.0 RR is inside the Region of Interest RI: start the election phase
    Step G3.2.1 RR checks its current state
    if Step G3.2.2RR is not in *Processing_IDReq*, nor in *Leader_IDReq*, nor in *Follower_IDReq* **then**
      Step G3.2.2.0 RR generates an *election_IDReq_msg*.
      Step G3.2.2.1 RR includes its distance to the origin of RI in the *election_IDReq_msg* and its knowledge about the requested service.
      Step G3.2.2.2 RR broadcasts the *election_IDReq_msg*.
      Step G3.2.2.3 RR becomes in the state *Processing_IDReq*.
    end if
  end if
end if

Destination	Location Information	Interface
Road component 1	(X1, Y1)	Interface number
Road component 2	(X2, Y2)	...
Road component n	(Xn, Yn)	...

(a) Routing table format

Service ID	Service Information
Service 1	Service Description
	LifeTime
	Service Location
	Candidate RI
	Other information
Service 2	...
Service ...	...

(b) Service table format

Figure 5.3: Formats of routing and service tables

- **Phase 2: Service Request Propagation:** The service request propagation phase is illustrated in Figure 5.2(b). A driver or passenger generates a request for a fixed service, a moving service, or a migratory service. In the service request, the user specifies the location where he wishes to find the service, i.e., the Region of Interest (RI). The region of interest is specified by a disc area represented by the coordinates of the origin of the disc and the radius from the fixed origin. The format of the service request or the Location-based Vehicular Reactive Discovery (L-VRD) packet is illustrated in Figure 5.1(b).

  The L-VRD packet contains both routing information (source address, destination address, sender address, packet ID, and time-to-live) and discovery information (packet type, requested service, service attributes, coordinates of the region of interest, and the distance from the current node to the center of the region of interest). The formal description of the service request propagation phase is elaborated in Algorithms 5.3.1 and 5.3.2 (Case B, C, F and G). The requester sends the service request L-VRD packet to the neighboring Roadside Routers and vehicles. The neighbors Roadside Routers and vehicles of the requesting vehicle receive the L-VRD packet and separate the service information from the routing information. The service information will be processed by the service module and the routing information will be handled by the routing module.

For service request propagation, we implement a *Location-aided Request Propagation mechanism* for an efficient request propagation.

The formal description of the *Location-aided Request Propagation mechanism* is elaborated in Algorithm 5.3.2 (Step G3.0 to Step G3.3). Every Roadside Router or vehicle that receives the service request or L-VRD packet determines through the location-aided request propagation mechanism whether it should forward the request or not in the following way. First, the Roadside Router or vehicle checks whether it is inside the Region of Interest or not. If not, then the Roadside Router or vehicle determines the distance that separates it from the origin of the region of interest. It compares this distance to the distance of the sending node received in the L-VRD packet. If the computed distance is shorter than the received distance, then the Roadside Router or vehicle forwards the request message. Otherwise, the request message is not forwarded. If the Roadside Router or vehicle is inside the region of interest, then the coming action depends from whether the current node is a RR or a vehicle. If the current node is a RR, then it forwards the service request and initiates a Leader Election process to elect a leader Roadside Router in the region of interest. The elected leader is the root of the computed spanning tree and is responsible for collecting local service replies from its children and generating the reply message that will be sent to the requesting vehicle. If the current node is a vehicle, then if it provides the requested service, then a service reply is sent to the requesting vehicle using the CLA-S [40]. Otherwise, it broadcasts the service request.

- **Phase 3: Leader Election and Service Reply Generation** The leader election and service reply propagation phase is executed only by Roadside Routers and is illustrated in Figures 5.2(c), 5.2(d), 5.2(e) and 5.2(f). A leader Roadside Router is elected in the region of interest. This leader is responsible for generating an aggregated service reply and its propagation to the service requester. The reason from electing a leader in the region of interest is to avoid sending many service replies to the service requester if there are many service providers of the same service inside the RI. The leader will send an aggregated reply message to the service requester. In the aggregated reply message, all service providers IDs and locations discovered inside the RI are added to the service reply message.

Algorithm 5.3.3 PROTOCOL RUNNING AT THE ROUTER RRI

```
Case H. receive a election_IDReq_msg message from Roadside Router RR_j:
if Step H0. RR is inside the RI then
    Step H0.0. RR checks its current state
    if Step H0.1. RR is not in Processing_IDReq, nor in Leader_IDReq, nor in Follower_IDReq then
        Step H0.1.0 RR generates an election_IDReq_msg.
        Step H0.1.1 RR includes its distance to the origin of RI in the election_IDReq_msg and its knowledge about the requested service.
        Step H0.1.2 RR broadcasts the election_IDReq_msg.
        Step H0.1.3 RR becomes in the state Processing_IDReq.
    end if
end if
Case I. Every election timer P (Generation of the spanning tree for IDReq):
Step I0. RR checks its current state.
if Step I1. current state is Processing_IDReq then
    if Step I1.0.RR has the minimum distance to the origin of RI then
        Step I1.0.0 RR becomes Leader_IDReq
    else
        Step I1.1 RR chooses its parent_IDReq which is its neighboring RRs that has the minimum distance to the origin of RI
        Step I1.1.0 RR becomes Follower_IDReq
        Step I1.1.1 RR sends its knowledge about the requested service to its parent in a local service reply message local_SRep
    end if
end if
Case J. RR is in the Leader_IDReq state and receives all local replies from its children
Step J0. RR aggregates the local replies and generates the SRep
Step J1. send the service reply message SRep
Case K. receives a service reply message SRep from Roadside Router RR_j:
if Step K0. RR_i is the corresponding router of the service requester then
    Step K0.0. send reply to requester vehicle.
else
    Step K1.0. the integrated module separates the service information from the routing information.
    Step K1.1. the routing module determines the forwarding zone to the destination.
    Step K1.2. The service module caches the service information in the service table.
    Step K1.3. send the reply message.
end if
```

The formal description of the election phase is elaborated in Algorithms 5.3.2 and 5.3.3 (from Step G3.2 to Step G3.3, Case H, I and J). A Roadside Router in the region of interest that receives a service request message starts the leader election process. It sets its current state to *processing_IDReq*. We piggyback the ID of the request to the processing state because a Roadside Router can participate in many location-based service discovery processes. Then, the Roadside Router generates a leader election message *election_msg_IDReq* that contains the following information: its ID, the distance to the origin of the region of interest, and the service provider of the requested service if its service table contains the requested service. The Roadside Router sends the *election_msg_IDReq* to its neighboring Roadside Routers. Any Roadside Router in the region of interest enters the *processing_IDReq* state if one of the following events happen: (i) it receives a service request with the ID *IDReq*, or (ii) it receives an *election_msg_IDReq*. All Roadside Routers in the *processing_IDReq* state wait for a predetermined period of time P then they decide whether to enter the *leader_IDReq* or the *follower_IDReq* states. A Roadside Router decides to enter the *leader_IDReq* state if and only if it has the minimum distance to the origin of the region of interest among its neighboring Roadside Routers. A Roadside Router decides to enter the *follower_IDReq* state if and only if at least one of its neighboring Roadside Routers has a shorter distance to the center of the region of interest than its distance to the center of the region of interest. A Roadside Router chooses its neighboring Roadside Router that has the minimum distance to the origin of the region of interest among its other neighbors as its parent (*parent_IDReq*) in the spanning tree.

At the end of the election process, a spanning tree including all of the Roadside Routers in the region of interest is generated. The root of this spanning tree is the elected leader *leader_IDReq*. After the construction of the spanning tree, the leaves send their replies to their corresponding parents. The local reply contains the service provider required by the service requester. The leader Roadside Router receives all the local replies from its children. After receiving all the local replies from its children. The leader generates an aggregated service reply message *SRep_msg* and sends it to the requesting vehicle. The service reply contains the IDs and locations of the service providers that satisfy the driver's or passenger's request. The service reply message includes the candidate region of interest for which the service providers have been selected. This information will be cached by intermediate Roadside Routers during the service reply propagation phase.

In our protocol, the election and the spanning tree construction processes are performed between Roadside Routers, which are assumed to be fixed and stable. Moreover, the election phase is executed for each service query. Thus, we do not need to maintain the elected leader and the constructed spanning tree after sending an aggregated service reply to the requester.

- **Phase 4: Service Reply Propagation:** The service reply propagation phase is illustrated in Figure 5.2(g). The formal description of the service reply generation and propagation phase is elaborated in Algorithms 5.3.1 and 5.3.3 (Case K and CC). In our EB-LocVSDP, an aggregated service reply that contains all the service providers of the requested service in the desired region of interest is sent by the elected RR. If a service provider detects the absence of a RR in its range, and it receives a service request that corresponds to its provided service, then it sends a service reply with its ID and location to the service requester. Otherwise, the elected leader is responsible for solving the discovery query with the collaboration of other Roadside Routers in the region of interest, and responsible for generating the aggregated reply message and sending it to the requester. For service providers data aggregation, we implement a simple mechanism. The service reply message contains a service reply table. Each time the leader RR receives a local reply from its neighboring RR, it adds the received service provider's information to its service reply table. Service providers are sorted so that the first record in the table corresponds to the closest service provider to the origin of the region of interest. After the reception of all local replies, the leader RR generates its unique aggregated service reply and sends it to the requesting vehicle using the CLA-S routing protocol [40]. The propagation of the aggregated service reply message from the leader RR or the reply message from the vehicle service provider is done through the $CLA-S$ [40] routing protocol. Since the service requester location information $(x0, y0)$ and average speed Vel_0 at the sending time $t0$ are sent in the service request message, the leader RR or the provider vehicle can determine the expected location zone of the service requester at time $t1$ when the service reply is ready to be sent. The expected zone of the service requester at the instant $t1$ would be the disc centered at the initial location of the requesting vehicle $(x0, y0)$ and having the radius $(t1 - t0)Vel\_0$. The aggregated service reply message or the reply message is sent towards the center of the expected zone of the requesting vehicle through the CLA-S routing protocol [40]. Once the aggregated service reply message or the reply message reaches the expected zone, it is broadcasted in order to make sure that it will be received by the requesting vehicle. Intermediate Roadside Routers or vehicles that receive the service reply message cache the service information in their corresponding service tables.

5.3.3 A Naive-LocVSDP Protocol

We propose a variant of the EB-LocVSDP by modifying the *Leader election and service reply generation phase* and the *Service reply propagation phase*. In Naive-LocVSDP, Roadside Routers in the region of interest, specified in the driver request, that receive a location-based service request send directly their service replies to the service requester. Every Roadside Router in the region of interest that receives the location-based service request checks its service information table. If the requested service exists in the service table, then a reply is sent back to the service requester with the provider information. In addition, if the service requested does not exist in the service information table, the Roadside Router sends a negative reply to the service requester.

5.4 Message Complexity and Cost Function of the EB-LocVSDP

We recall that in our system model, we consider a rectangular VANet of area $a \times b$, where a and b are the length and the width of the VANet respectively. The VANet is composed of straight lanes and perpendicular lanes where many service providers are deployed as shown in Figure 5.2(a). Roadside Routers are randomly distributed in the VANet around the known services providers, the predictable congested area, and the predetermined intermittent areas. Roadside routers in the communication range of each other are clustered together. They

Table 5.1: Variables description

Variable name	**Variable description**
SP	service provider
VRi	service requester i
RI	a region of interest
$Oi(x_{Oi}, y_{Oi})$	origin coordinate of RI of the i^{th} request
R_{RIi}	radius of RI of the i^{th} request
RR	Roadside Router
$VRi(x_{VRi}, y_{VRi})$	the coordinates of VRi
Vel_{VRi}	velocity of the service requester VRi
$t0i$	the instant of sending the service request from VRi
$t1i$	the instant of sending the service reply from the RR to VRi
nSP	number of SPs in VANet
$nReqs$	number of service requests during T
λ_{adv}_i	advertisement period corresponding to SP i
T	the simulation duration
ρ	the density of VANet in terms of RRs
ρ_V	the density of VANet in terms of Vehicles
$Area\_RI$	the area of RI
L-VRD	Location-based Vehicular Reactive Discovery
L-VPD	Location-based Vehicular Proactive Discovery
$SRep\_msg$	Aggregate service reply message
$\lambda_{propagt}$	the propagation time of a message in VANet
λ_{proct}	the processing time for one election phase
$size\_Adv$	size of an advertisement message
$size\_Req$	size of a request message = 40 bits
$size\_Elec$	size of an election message = 20 bits
$size\_LRep$	size of a local reply message = 20 bits
$size\_Rep$	size of a reply message = 40 bits
$size\_Adv$	size of an advertisement message = 20 bits
r	transmission range of a vehicle

are fixed and stable entities. Every Roadside Router knows its location coordinates and is equipped with two interfaces tuned to two different channels. One interface is used for communication between Roadside Routers, and the other interface is used for communication to vehicles or to other Roadside Routers in order to permit sending and receiving messages at the same time. Participating vehicles are equipped with one interface used for communication to other vehicles or to Roadside Routers. In the following, we present the message complexity computation and the cost function of our EB-LocVSDP. We use the parameters described in Table 5.1.

5.4.1 Message Complexity

The message complexity of our EB-LocVSDP consists of the number of advertisement messages, the number of location based requests, the number of election messages, the number of local reply messages, and the number of reply messages.

Lemma 5.4.1. *The number of advertisement messages during the simulation time is:*

$$nTotal\_Adv = \sum_{k=1}^{nSP}(\frac{T}{\lambda_{adv}\_k}) \quad (5.1)$$

Proof: The computation of the number of advertisement messages is the result of the periodic advertisement of packets needed to inform the neighboring Roadside Routers about an existing service. Every service provider sends an advertisement message every λ_{adv} period. Consequently, the total number of advertisement messages sent by one service provider during the period T is $n\_VPD = \frac{T}{\lambda_{adv}}$. The overall number of advertisement messages of our $EB - LocVSDP$ depends on the number of service providers in the VANet and the period of advertisement of each provider. Hence, the total overhead in the VANet for all the service providers during the simulation period T is given by:

$$nTotal\_VPD = \sum_{k=1}^{nSP}(\frac{T}{\lambda_{adv}\_k}) \quad (5.2)$$

Lemma 5.4.2. *The total number of location-based request messages $n\_Total\_Loc\_Req$ for $nReqs$ service requests during the simulation period T is:*

$$n\_Total\_Loc\_Req = \sum_{i=1}^{i=nReqs} ((((x_{Oi} - x_{VRi}) - R_{RIi}) \times (2 \times R_{RIi})) \times (\rho + \rho_V)) \quad (5.3)$$

Proof: In our EB-LocVSDP, location-based service request messages are propagated in a controlled manner as explained in the *Service request propagation phase* of our algorithm. In this phase, only Roadside Routers and vehicles that are closer to the region of interest RI forward the location-based service request. In order to prevent redundant request propagation, the same location-based service request is forwarded only once by the same Roadside Router or vehicle. Thus, the number of location-based service requests for a driver's or passenger's query is equal to the number of Roadside Routers and vehicles in the location-based propagation area. The location-based request propagation area has the form of a rectangle as depicted in Figure 5.2(b). Let us denote the area of this rectangle by $Area\_Rectangle$, the area of propagation of the location-based service request by $Area\_LocRequest$, and the area of the region of interest RI defined as the disc $D(O, R_{RI})$ by $Area\_RI$. We depict from Figure 5.2(b) that $Area\_LocRequest$ is approximately equal to $Area\_Rectangle$. If we know the densities ρ and ρ_V in terms of Roadside Routers and vehicles respectively in the VANet, then we can determine the number of forwarding Roadside Routers and vehicles $n\_F\_Area\_LocRequest$ inside $Area\_LocRequest$. The number of location-based request messages n_Loc_Req for one service request is:

$$\begin{aligned} n\_Loc\_Req &= n\_F\_Area\_LocRequest \\ &= Area\_LocRequest \times (\rho + \rho_V) \\ &= Area\_Rectangle \times (\rho + \rho_V) \\ &= (((x_O - x_{VR}) - R_{RI}) \times (2 \times R_{RI})) \times (\rho + \rho_V) \end{aligned}$$

The total number of location-based request messages $n\_Total\_Loc\_Req$ for $nReqs$ service requests is:

$$n\_Total\_Loc\_Req = \sum_{i=1}^{i=nReqs} ((((x_{Oi} - x_{VRi}) - R_{RIi}) \times (2 \times R_{RIi})) \times (\rho + \rho_V)) \tag{5.4}$$

Lemma 5.4.3. *The total number of election messages $n\_Total\_election\_msg$ for nReqs service requests is:*

$$n\_Total\_election\_msg = \sum_{i=1}^{i=nReqs} (\Pi \times R_{RIi}^2 \times \rho) \tag{5.5}$$

Proof: In our EB-LocVSDP, only RRs participate in the election phase. Every RR in the region of interest sends the election message once for a specific service query. Thus, the number $n\_election\_msg$ of election messages for one query is equal to the number of Roadside Routers $n\_RR\_Area\_RI$ in the region of interest RI:

$$\begin{aligned} n\_election\_msg &= n\_RR\_Area\_RI \\ &= Area\_RI \times \rho \\ &= \Pi \times R_{RI}^2 \times \rho \end{aligned}$$

The total number of election messages $n\_Total\_election\_msg$ for $nReqs$ service requests is:

$$n\_Total\_election\_msg = \sum_{i=1}^{i=nReqs} (\Pi \times R_{RIi}^2 \times \rho) \quad (5.6)$$

Lemma 5.4.4. *The total number of local reply messages* $n\_Total\_localRep\_msg$ *for* $nReqs$ *service requests is:*

$$n\_Total\_localRep\_msg = \sum_{i=1}^{i=nReqs} (\Pi \times R_{RIi}^2 \times \rho) \quad (5.7)$$

Proof: In our EB-LocVSDP and during the local-reply messages propagation, every Roadside Router in the region of interest sends its local reply message to its parent in the spanning tree generated for a specific service request. Thus the number $n\_localRep\_msg$ of local reply messages for one query is equal to the number of Roadside Routers $n\_RR\_Area\_RI$ in the region of interest RI:

$$\begin{aligned} n\_localRep\_msg &= n\_RR\_Area\_RI \\ &= Area\_RI \times \rho \\ &= \Pi \times R_{RI}^2 \times \rho \end{aligned}$$

The total number of local reply messages $n\_Total\_localRep\_msg$ for $nReqs$ service requests is:

$$n\_Total\_localRep\_msg = \sum_{i=1}^{i=nReqs} (\Pi \times R_{RIi}^2 \times \rho) \tag{5.8}$$

Lemma 5.4.5. *The total number of reply messages $n\_Total\_Rep\_msg$ for nReqs service requests during the simulation period T is:*

$$n\_Total\_Loc\_Rep =$$

$$\sum_{i=1}^{i=nReqs} ((((x_{Oi} - x_{VRi}) + R_{RIi} + (Vel_{VRi} \times (t1i - t0i))) \times (2 \times (Vel_{VRi} \times (t1i - t0i)))) \times (\rho + \rho_V)) \tag{5.9}$$

Proof: In our EB-LocVSDP, if a requested service provider has no RR in its range, then it sends a reply message to the requester. However, if the service provider is in the communication range of a RR, then the elected leader Roadside Router during a service transaction collects local replies from all Roadside Routers in the region of interest RI and aggregates them in one service reply. After this, it sends a unique service reply message to the requesting vehicle with the different service providers of the same service. The service reply is geocasted to the service requester from the leader Roadside Router. We denote $Area\_LocReply$ the propagation area of the reply message. In order to determine $Area\_LocReply$, the leader RR determine the expected zone of the service requester at the current time $t1$. Since the service request carries the location information (x_{VR}, y_{VR}) and the velocity (Vel_{VR}) of the requesting vehicle VR at time $t0$, the expected zone area of the service requester is determined as follow:

$$Exp_{Area} = \Pi \times (Vel_{VR} \times (t1 - t0))^2 \tag{5.10}$$

Thus, the number of forwarded reply messages for a driver's or passenger's query is equal to the number of Roadside Routers and vehicles in the reply propagation area. The reply propagation area has the form of a rectangle as depicted in Figure 5.2(g). Let us denote the area of this rectangle by $RepArea\_Rectangle$, the area of propagation of the reply message by

Area_LocReply, and the expected zone area of the service requester at time $t1$ by Exp_{Area}. We depict from Figure 5.2(g) that *Area_LocReply* is approximately equal to *RepArea_Rectangle* that includes the expected zone area and part of the region of interest area. Knowing the densities ρ and ρ_V in terms of Roadside Routers and vehicles respectively in the VANet, we determine the number of forwarding Roadside Routers and vehicles *n_F_Area_LocReply* inside *Area_LocReply*. The number of forwarded reply messages *n_Loc_Rep* for one service request is:

$$\begin{aligned} n\_Loc\_Rep &= n\_F\_Area\_LocReply \\ &= Area\_LocReply \times (\rho + \rho_V) \\ &= RepArea\_Rectangle \times (\rho + \rho_V) \\ &= (((x_{VR} - x_O) + R_{RI} + (Vel_{VR} \times (t1 - t0))) \times (2 \times (Vel_{VR} \times (t1 - t0)))) \times (\rho + \rho_V) \end{aligned}$$

The total number of reply messages *n_Total_Loc_Rep* for *nReqs* service requests is:

$$n\_Total\_Loc\_Rep =$$

$$\sum_{i=1}^{i=nReqs} ((((x_{VRi} - x_{Oi}) + R_{RIi} + (Vel_{VRi} \times (t1i - t0i))) \times (2 \times (Vel_{VRi} \times (t1i - t0i)))) \times (\rho + \rho_V)) \quad (5.11)$$

Lemma 5.4.6. *The total number of messages in our EB-LocVSDP during the simulation period T is:* $MC\_LocVSDP = \sum_{k=1}^{nSP}(\lambda_{adv}\_k \times T) + \sum_{i=1}^{i=nReqs}((((x_{Oi} - x_{VRi}) - R_{RIi}) \times (2 \times R_{RIi})) + (((x_{VRi} - x_{Oi}) + R_{RIi} + (Vel_{VRi} \times (t1i - t0i))) \times (2 \times (Vel_{VRi} \times (t1i - t0i))))) \times (\rho + \rho_V) + 2 \times (\Pi \times R_{RIi}^2 \times \rho)$

Proof: The total number of messages *MC_LocVSDP* of our EB-LocVSDP during the simulation period *T* comprises: the total number of advertisement messages, the total number of location-based request messages, the total number of election messages, the total number of local reply messages and the total number of reply messages. The Lemma 5.4.6 is deduced from Lemmas 5.4.1, 5.4.2, 5.4.3, 5.4.4 and 5.4.5.

5.4.2 EB-LocVSDP Cost Function

For the estimation of the cost function of EB-LocVSDP, we assume that vehicles are distributed uniformly in the VANet. For reason of simplification, we use a uniform distribution of vehicles. However, our protocol can be applied to any distribution type. In [12], Bettstetter *et al.* discussed the probability density function (pdf) of a random variable x that denotes a straight line distance between two random nodes in a $a \times b$ rectangular topology, where a and b are the length and the width of the VANet respectively:

$$P_X(x) = \frac{4x}{a^2b^2}(\frac{\Pi}{2}ab - ax - bx + 0.5x^2) \tag{5.12}$$

Then, they compute the expected distance $E[x]$ between two nodes as:

$$E[x] = 0.521a \tag{5.13}$$

Knowing the transmission range r of a wireless vehicle, the expected number of hops $E[h]$ between a service provider and a service requester is given by:

$$E[h] = E[x]/r \tag{5.14}$$

We assume that the expected distance between a service provider and a service requester is equal to $0.521a$. In the following, we assume that the expected distance between a requesting vehicle VRi and the origin of the RI, denoted by Oi, is equal approximately to $0.521a$.
For the theoretical estimation of EB-LocVSDP cost functions, we use the values in Table 5.2. The cost of service advertisement messages $Cost\_Adv$ is :

$$\begin{aligned} Cost\_Adv &= n\_Total\_Adv \times size\_Adv \\ &= nSP \times (\frac{T}{\lambda_{adv}\_k}) \times size\_Adv \\ &= nSP \times A \end{aligned}$$

where $A = (\frac{T}{\lambda_{adv}\_k}) \times size\_Adv$.

The cost of service requests messages $Cost\_Reqs$ is :

$$
\begin{aligned}
Cost\_Reqs &= n\_Total\_Loc\_Req \times size\_Reqs \\
&= nReqs \times ((((x_{Oi} - x_{VRi}) - R_{RIi}) \times (2 \times R_{RIi})) \times (\rho + \rho_V)) \times size\_Reqs \\
&= nReqs \times B
\end{aligned}
$$

where $B = \times((((x_{Oi} - x_{VRi}) - R_{RIi}) \times (2 \times R_{RIi})) \times (\rho + \rho_V)) \times size\_Reqs$.

$$
\begin{aligned}
Cost\_Elec &= nReqs \times n\_Total\_election\_msg \times size\_Elec \\
&= nReqs \times (\Pi \times R_{RIi}^2 \times \rho) \times size\_Elec \\
&= nReqs \times C
\end{aligned}
$$

where $C = (\Pi \times R_{RIi}^2 \times \rho) \times size\_Elec$.

$$
\begin{aligned}
Cost\_LRep &= nReqs \times n\_Total\_localRep\_msg \times size\_LRep \\
&= nReqs \times (\Pi \times R_{RIi}^2 \times \rho) \times size\_LRep \\
&= nReqs \times D
\end{aligned}
$$

where $D = (\Pi \times R_{RIi}^2 \times \rho) \times size\_LRep$.

$$
\begin{aligned}
Cost_{Rep} &= nReqs \times n\_Total\_Loc\_Rep \times size_{Rep} \\
&= nReqs \times ((((x_{VRi} - x_{Oi}) + R_{RIi} + (Vel_{VRi} \times (t1i - t0i))) \times (2 \times (Vel_{VRi} \times (t1i - t0i)))) \times \\
&\quad (\rho + \rho_V)) \times size_{Rep} \\
&= nReqs \times E
\end{aligned}
$$

where $E = ((((x_{VRi} - x_{Oi}) + R_{RIi} + (Vel_{VRi} \times (t1i - t0i))) \times (2 \times (Vel_{VRi} \times (t1i - t0i)))) \times (\rho + \rho_V)) \times size_{Rep}$.

$$\begin{aligned} Cost_{LocVSDP} &= Cost\_Reqs + Cost\_Elec + Cost\_LReps + Cost\_Reps \\ &= nReqs \times (B + C + D + E) + nSP \times A \end{aligned}$$

Assuming that 60% of the service requests are answered from the cached information in the service tables,

$$Cost_{LocVSDP} = 0.4 \times (nReqs \times (B + C + D + E)) + nSP \times A \quad (5.15)$$

Thus, EB-LocVSDP is $O(nReqs)$. This means that our protocol scales linearly with the number of requests.

5.5 Performance Evaluation of LocVSDPs

In this section, we present the performance evaluation of the two versions of our proposed location-based service discovery protocol (LocVSDP): *(i) the naive-LocVSDP* and *(ii) the Election-Based LocVSDP*. Then, we present the comparison results of our protocols to the VITP [25] in a realistic traffic pattern. We choose the VITP for comparison with our protocols, because VITP is a location-based service discovery protocol dedicated to vehicular networks.

In the following, we recall the functionalities of the VITP and we explain the adaptation that we made to VITP to make it suitable for comparison with our discovery protocols. A VITP transaction consists of the following phases: the first one is a *query phase*. In this phase, the request is forwarded through intermediate vehicles to the targeted zone using a geographic routing. The intermediate vehicles could be non VITP-enabled. When the query reaches its targeted zone and is received by a VITP peer, the second phase, the *computation phase*, starts. In this phase, a Virtual Ad-Hoc Server (VAHS) is formed by the VITP peers to resolve the query. When a VAHS peer detects a predefined *Return Condition*, it generates a reply and sends it through the VANet to the source road segment from which the query was initiated using a geographic routing. This phase is the *reply phase*. In our simulation experiments, we varied the return condition from 1 to 10 service providers of the same service. As soon as the reply reaches the source road segment, VITP enters the last phase, which is the *reply delivery phase*. In this phase, the reply message is broadcasted in the source segment. In the VITP, the road is supposed to be divided into segments. The region of interest in VITP is defined as the coordinates of a starting point and a predefined distance from the starting point.

In our LocVSDPs, we define the region of interest as the coordinates of the center of the region of interest and a radius from the center point. In our simulation experiments, we adapted the region of interest of VITP as the area defined by the coordinates of the center of the region of interest and a radius from the center point as in our LocVSDPs. The source segment is adapted also to a circular area. We conducted an extensive set of experiments in the network simulator NS2 [4]. In our experiments, we varied the number of service requests from 100 to 1000 to study the scalability of the compared protocols. We choose to evaluate the LocVSDP's performance using mobility traces that correspond to realistic road-traffic pattern: *Manhattan* traffic pattern. In this traffic pattern, vehicles move in a Manhattan model and vehicles use the LocVSDPs for location-based service discovery. We compare our algorithms to the VITP and discuss the obtained results.

In our simulations, we model the arrival process to the vehicular network as a poisson process with an arrival rate (λ). We refer to the number of vehicles per meter square in the network as its density in terms of vehicles (*Density*). A road section has a fixed length and surface, and has the same density than the vehicular network. We consider Vel the average speed of a vehicle. Assuming N the random variable that defines the number of vehicles in a road section of length x, the probability that there are n vehicles in a road section of length (x) is:

$$P[N(x) = n] = \frac{(\frac{x}{Vel}\lambda)^n}{n!} e^{-\lambda \frac{x}{Vel}} \tag{5.16}$$

In our experiments, we want to prove that our LocVSDPs are scalable in a highly dense vehicular network and with the increasing number of requests. For this purpose, we assume that the network density is 3.34 $10^{-4} vehicles/m^2$. Besides, we assume that vehicles move with the average speed Vel equal to $20m/s$. We base our assumptions on the traffic flow state for different densities in [55] and we use the service level D, where the network is uncongested but borders are unstable. We set the simulation time to 1200 seconds. Thus, we deduce the arrival rate (λ) for our simulations as ($\lambda = \lambda(x) = \frac{\overline{N(x)}Vel.b}{x}$), where $\overline{N(x)} = \sum_{n=1}^{\infty} nP[N(x) = n]$.

In our experiments, we use the IEEE 802.11 as wireless medium with a data transmission rate of 11Mbps and a transmission range of 200 meters. The received signal strength threshold is set to 200 meters. We use the Connection Less Approach for Streets (CLA-S) [40] as routing protocol for communication between Roadside Routers in the wireless VANet and for communication between Roadside Routers and vehicles. In our simulations, we consider 10 service providers publishing the same service and 40 drivers or passengers service clients requesting the same service. When a vehicle enters in the VANet, the driver or the passenger chooses a random time to initiate its request.

Simulation Experiments

In this section, we report on simulation experiments and compare our algorithms to the VITP using a Manhattan traffic model. We use the simulation environment illustrated in Table 5.2. In the course of our experiments, we define the request number as the number of service queries sent by vehicle clients. We choose to evaluate the performance of our algorithm using the following metrics:

Table 5.2: Simulation environment

Parameter name	Parameter value
Wireless medium	IEEE 802.11
Data transmission rate	11Mbps
Transmission range (meters)	200
Received signal strength threshold (meters)	200
Average vehicle's Speed (meter/second)	20
Simulation time(seconds)	1200
Simulation area ($meter^2$)	$600 \times 600 = 360,000$
Average vehicles density (vehicles/Km^2)	334
Average RRs density in a cluster(RR/Km^2)	40
Clients' number	40
servers' number	10
Region of interest surface ($meter^2$)	$\Pi \times 100^2$

- *Success rate* which indicates the average fraction of successful service queries over the total number of service queries sent in the VANet during the simulation time;

- *Average response time* which defines the average time of successful request transactions. It measures the elapsed time for getting a valid service reply in response to a service request sent by a vehicle. This metric takes into account several factors such as transmission and message processing delay, just to mention a few;

- *Bandwidth usage* which measures the bandwidth needed to satisfy the driver's or passenger's service requests; and

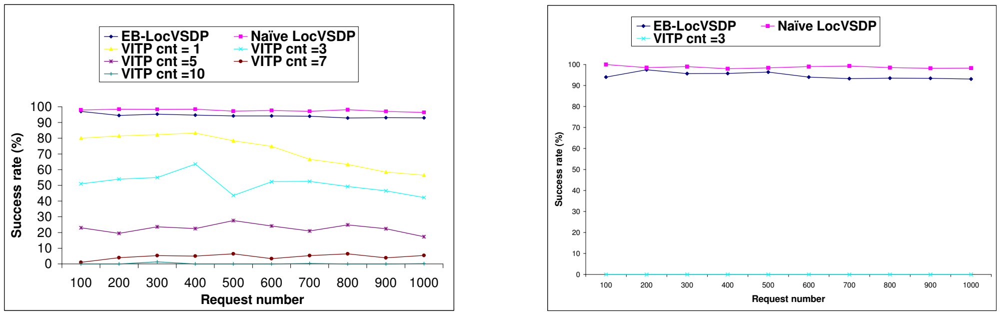
- *Bandwidth efficiency* which calculates the percentage of bandwidth used for successful transactions.

In the course of our experiments, we varied the number of service requests between 100 and 1000. We set the simulation time to 1200 seconds with a warm up period of 120 seconds. Service requests initial time is generated randomly after the first 120 seconds. The following experimental data were obtained by averaging several runs with a confidence of 95 percent. We used different seeds for every run. The results are summarized in graphs 5.4, 5.5, 5.6, 5.7 and 5.8.

Let us now turn to our results. Figure 5.4 portrays the success rate of our naive and election-based LocVSDPs, and the VITP with different return conditions. Figure 5.4(a) illustrates the success rate of the compared protocols when 10 service providers providing the requested service exist in the region of interest specified in the location-based service request queries. Figure 5.4(b) describes the success rate of the compared protocols when none of the service providers of the requested service exist in the region of interest specified in the driver's location-based request. As we can see, our LocVSDPs outperform considerably the VITP in terms of success rate. The naive-LocVSDP achieves almost 97 percent as a success rate in the discovery of at least 90 percent of the service providers in the requested region of interest. The election-based LocVSDP (EB-LocVSDP) succeeds as well to find at least 90 percent of the service providers in the specified region of interest with a rate that exceeds 95 percent when the number of requests increases from 100 to 1000. Even when there is no service provider in the requested region of interest, our LocVSDPs succeed in returning a reply to the service requester with a rate greater than 95 percent. In the returned reply message there is an indication that the requested service does not exist in the specified region of interest. However, the VITP achieves lower success rate than the LocVSDPs even when the return condition is set to one service provider. Thus, the VITP achieves lower than 80 percent as a success rate in the discovery of only 10 percent of service providers of the requested service in the desired region of interest.

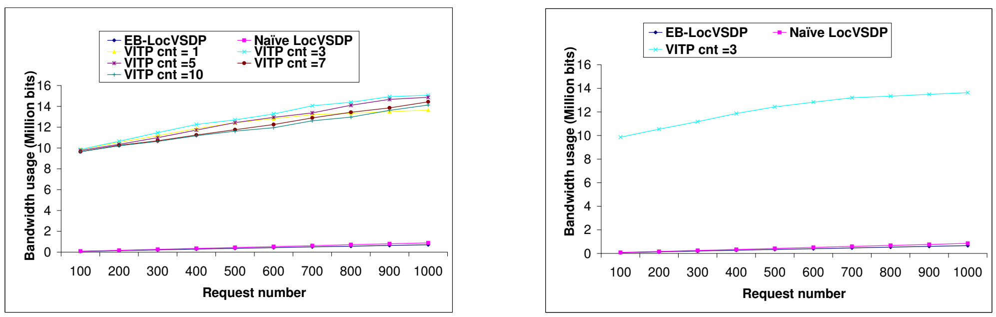
The success rate of VITP decreases considerably with the increase of the return condition. As can be seen in the graphs in Figure 5.4(a), VITP achieves a success rate lower than 50 percent for the discovery of 30 percent of service providers in the desired region of interest. Besides, its success rate is lower than 25 percent for the discovery of 50 percent of the service providers, lower than 5 percent for the discovery of 70 percent of the service providers, and almost 0 percent for the discovery of 100 percent of the service providers when the number of requests increases from 100 to 1000.

If there is no service provider in the desired region of interest, the success rate of VITP in returning a reply is 0 percent as shown in Figure 5.4(b). This is mainly due to the fact that the VITP does not return a reply as long as the return condition is not satisfied. In this case,



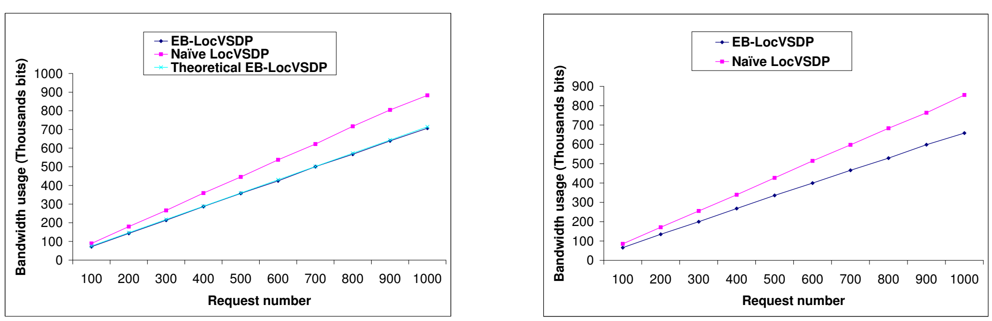
(a) The requested service exists (b) The requested service does not exist

Figure 5.4: Success Rate comparison of LocVSDPs to VITP in a City traffic scenario



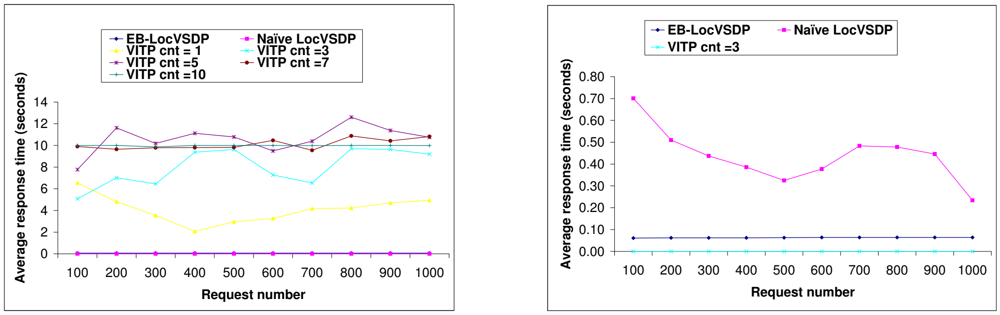
(a) The requested service exists (b) The requested service does not exist

Figure 5.5: Bandwidth usage comparison of LocVSDPs to VITP in a City traffic scenario



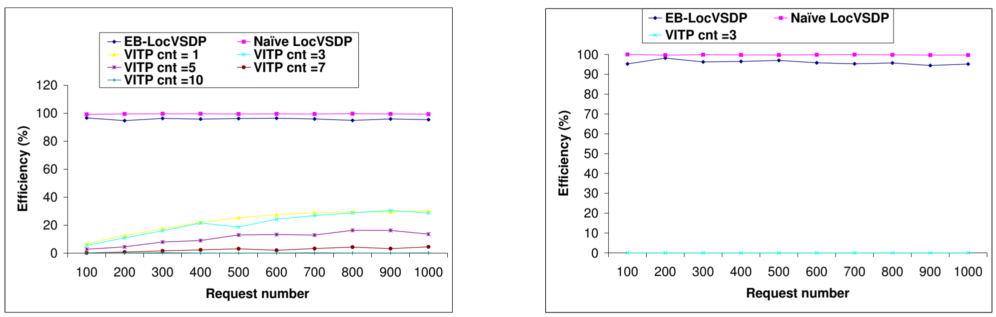
(a) The requested service exists (b) The requested service does not exist

Figure 5.6: Bandwidth usage comparison of LocVSDPs in a City traffic scenario



(a) The requested service exists (b) The requested service does not exist

Figure 5.7: Average response time comparison of LocVSDPs to VITP in a City traffic scenario



(a) The requested service exists (b) The requested service does not exist

Figure 5.8: Bandwidth efficiency comparison of LocVSDPs to VITP in a City traffic scenario

the requester waits for 10 seconds until the request times out to deduce that the service does not exist in the specified region or that the request or reply was dropped. Thus, the VITP client could initiate another request for the same service in the same region, because it is not sure yet that the service does not exist in the specified region. However, in our protocol a negative reply is returned to the service requester if the service does not exist in the region of interest, confirming hence to the service requester that the requested service does not exist in the specified region of interest.

The main reason behind the high success rate in our election-based LocVSDP is mainly due to the fact that a unique service reply is generated by the elected leader in the desired region of interest to be returned to the requesting vehicle. Moreover, we will prove later that the average response time for a service transaction is very low (in the order of tens of milliseconds) which helps to decrease the dropping of reply messages even in a highly mobile environment. The main reason behind the high success rate of the naive-LocVSDP is that every router in the region of interest returns a reply to the service requester. The reply could contain either the service provider information or an indication that the Roadside Router does not have the requested service in its service table.

Figure 5.5 portrays the bandwidth usage of our LocVSDPs and the VITP with different return conditions. Figure 5.6 emphasizes on the graphs of the EB-LocVSDP and the naive-LocVSDP. Figures 5.5(a) and 5.6(a) illustrate the bandwidth usage of the compared protocols when 10 different service providers of the requested service exist in the desired region of interest. Figure 5.6(a) presents as well the curve related to the theoretical computation of the bandwidth usage of the EB-LocVSDP.

Figures 5.5(b) and 5.6(b) portray the bandwidth usage of the compared protocols when there is no service provider that provides the requested service in the specified region of interest. We notice from Figures 5.5(a) and 5.5(b) that our LocVSDPs outperform greatly the VITP with different return conditions in terms of bandwidth usage. We notice also that for all protocols, the bandwidth usage increases with the increase of the number of requests. The bandwidth usage of our EB-LocVSDP is in the order of 71 Kbits for 100 requests as depicted from Figure 5.6(a). It increases with about 70 Kbits every time the number of requests increases by 100. It reaches around 700 Kbits for 1000 requests. The curve related to the theoretical computation of the bandwidth usage of EB-LocVSDP shows almost the same behavior than the curve resulting from the simulations.

For the naive-LocVSDP, the bandwidth usage is less than 90 Kbits for 100 requests. It increases with around 85 Kbits, each time the number of requests increases by 100. It reaches more than 880 Kbits for 1000 requests. Thus, the EB-LocVSDP achieves 25 percent less bandwidth usage than the naive-LocVSDP. This is mainly due to the fact that our EB-LocVSDP generates and sends a unique reply to the service requester. Whereas, in the naive-LocVSDP every Roadside Router in the region of interest sends a reply to the service requester as soon as it receives the location-based request. We notice also that the increase in the bandwidth usage in both LocVSDP versions is almost linear because we assume in our simulation experiments that the location and the area of the region of interest are the same for all service requests. Thus, the increase in the bandwidth usage is mainly due to the increase of the number of location-based service requests and their corresponding service replies.

We notice from the LocVSDPs curves in Figures 5.6(a) and 5.6(b) that the behavior of the election-based LocVSDP curves and the behavior of the naive-LocVSDP curves are the same whether or not a service provider exists in the desired region of interest. This is mainly due to the fact that the election-based LocVSDP generates a unique spanning tree, elects a unique leader, and sends a unique reply to the service requester regardless of the existence of a service provider in the requested region of interest. In the naive version of LocVSDP, every Roadside Router in the region of interest sends a reply to the service requester to inform him about its knowledge of the existence or not of a service provider of the requested service in the desired region of interest.

For the VITP, we notice from Figure 5.5(a) that the bandwidth usage decreases with the increase of the return condition when 10 service providers exist in the requested region of interest. This decrease in the bandwidth usage is not due to the good performance of the VITP with the increase of the return condition but it is due to the bad performance of VITP when we increase the return condition. In fact, we have shown in Figure 5.4(a) that the success rate of VITP decreases drastically with the increase of the return condition to reach almost 0 percent for a return condition equal to 10. Consequently, the reason behind the decrease of the bandwidth usage of VITP with the increase of the return condition is that VITP blocks in its second phase (the VAHS computation phase) when the return condition increases and becomes hard to satisfy. This explains the low success rate of VITP with the increase of the return condition. In addition, when VITP blocks in its second phase (VAHS computation), the third phase (reply dispatch) and the forth phase (reply delivery) are not accomplished. This explains the decrease of the bandwidth usage when the success rate decreases, which is related to the increase of the return condition in VITP.

As observed in Figures 5.5(a) and 5.5(b), the bandwidth usage of VITP with different return conditions is considerably higher than our LocVSDPs when 10 service providers and 0 service providers exist in the region of interest respectively. For example, the bandwidth used by the VITP with a return condition equal to 3 is 20 times larger than the bandwidth of the election-based LocVSDP, and 17 times greater than the bandwidth of the naive-LocVSDP. There are many weaknesses and reasons that explain the high bandwidth usage in VITP. First, vehicles in the VITP have to discover their neighbors for the good functioning of the algorithm. Thus, every vehicle has to send a neighbor discovery message every one second, which incurs a useless overhead in the VANet. In our LocVSDPs, there is no need for neighbor discovery for the good functioning of our protocols.

Second, the VITP does not rely on any infrastructure to reduce the number of messages exchanged in large-scale VANets. Thus, during the query phase and the reply phase in VITP, a large number of messages is exchanged that increases the bandwidth usage in the VANet. As a contrast, our LocVSDPs rely on a cluster-based infrastructure comprising Roadside Routers that function as directories and reduce the number of messages exchanged in the VANet.

Third, the computation phase in VITP is not deterministic, and during the computation of a reply, a vehicle in the region of interest may receive the same computation message many times. However, in our EB-LocVSDP the construction of a spanning tree and the election of a unique leader break the ties and permit to generate a unique reply in a short time and with less messages exchanged. Moreover, in our naive-LocVSDP every Roadside Router sends its reply to the requester. Since the number of Roadside Routers is much lower than the number of vehicles in the VANet, our naive-LocVSDP does not incur so much overhead. The low response time of a successful query in our protocol, in addition to the usage of the CLA-S [40] routing protocol, guarantee the reception of the reply by the requester even in a highly mobile environment. Besides, we notice from the curves of VITP when the return condition is equal to 3 in Figures 5.5(a) and 5.5(b), that the bandwidth usage for 1000 requests when the requested service exists in the region of interest is around 5 percent higher than the bandwidth usage when the requested service does not exist in the desired region of interest. As we explained earlier, this is mainly related to the success rate which is 0 percent in Figure 5.4(b) when the service does not exist in the region of interest and more than 40 percent when the service exists in the region of interest as shown in Figure 5.4(a). If the success rate is 0, the VITP blocks in the computation phase and avoids the overhead incurred in the reply dispatch phase and the reply delivery phase. To conclude, the bandwidth usage of our LocVSDPs outperforms considerably the VITP even though the return condition of VITP is only 10 percent of the existing service providers in the desired region of interest, and that our protocol returns 90 percent of the existing services in the desired region of interest.

Figure 5.7 illustrates the average response time of our LocVSDPs and the VITP with different return conditions. Figure 5.7(a) portrays the average response time of the compared protocols when 10 different service providers of the requested service exist in the desired region of interest. Figure 5.7(b) illustrates the average response time of the compared protocols when there is no service provider that provides the requested service in the specified region of interest.

Figure 5.7(a) shows that the average response time of service discovery transactions in our election-based LocVSDP is in the order of 65 milliseconds for a number of requests ranging from 100 to 1000 when 10 service providers exist in the desired region of interest and when our EB-LocVSDP returns at least 90 percent of the service providers information in the region of interest. If there is no service provider that satisfies the driver's request in the desired region of interest, the average response time of successful transactions in our election-based LocVSDP is also in the order of 65 milliseconds as shown in Figure 5.7(b). Consequently, in the EB-LocVSDP version, our protocol achieves a reasonable and good average response time (65 milliseconds) for successful service discovery queries transactions.

The naive version of LocVSDP achieves around 10 milliseconds as average response time for a number of requests ranging from 100 to 1000 when 10 service providers exist in the desired region of interest as shown in Figure 5.7(a). However, the average response time in naive-LocVSDP is less than 1 second when there is no service provider that satisfies the driver's request in the desired region of interest as shown in Figure 5.7(a). We notice that there is a difference between the average response time of the naive-LocVSDP in the presence of service providers in the requested region of interest, and in their absence. This is due to the fact that in the naive-LocVSDP, the response time of a successful transaction is the period elapsed between the time when the service request has been sent and the first positive reply that is received that contains information about the requested service provider. Thus, when there is at least one service provider in the region of interest, the requester will receive a service reply from the corresponding Roadside Router as soon as the request reaches the region of interest. However, when there is no service provider in the region of interest, the service requester will receive negative service replies from Roadside Routers in the region of interest. In this case, we compute the response time as the period elapsed between the time when the service request has been sent and the last negative reply that is received.

Moreover, we notice a difference between the average response time of the EB-LocVSDP, which is in the order of 65 milliseconds, and the average response time of the naive-LocVSDP, which is in the order of 10 milliseconds, when 10 service providers exist in the desired region of interest as shown in Figure 5.7(a). This is mainly due to the fact that for the naive version of LocVSDP, the average response time is computed for the first positive reply received by the service requester. However, for the election based version of LocVSDP, the average response time is computed for a unique aggregated reply message. This latter is sent by the elected leader and contains at least 90 percent of the existing service provider's information in the requested region of interest. Moreover, in the EB-LocVSDP and during the election phase, a timer is used to construct the spanning tree. Then leaf nodes in the spanning tree have a waiting time before they send the local service replies to their parents to guarantee correctness of our EB-LocVSDP. Thus, a waiting period is necessary during the election and local reply propagation phase of our EB-LocVSDP. This explains the difference in the average response time between the EB-LocVSDP (order of 65 milliseconds) and the naive-LocVSDP (order of 10 milliseconds) when 10 service providers exist in the desired region of interest.

As we noticed earlier, the average response time in naive-LocVSDP is less than 1 second when there is no service provider that satisfies the driver's request in the desired region of interest as shown in Figure 5.7(a). Since in this case the response time is computed as the period elapsed between the time when the service request has been sent and the last negative reply that is received, we infer that less than 1 second is required in order to receive a complete response that states whether or not a service provider exists in the desired region of interest. We notice from Figure 5.7(a) that the average response time for a query transaction in the VITP increases with the increase of the return condition for a return condition between 10 and 100 percent of the available service providers in the region of interest and for a number of requests ranging from 100 to 1000. From Figure 5.7(b), we notice that the average response time for VITP is 0. This is because if there is no service providers in the desired region of interest, the success rate is 0 for VITP as depicted in Figure 5.4(b). In summary, the average response time of our LocVSDPs outperform the VITP. It is in the order of tens of milliseconds in our protocols and in the order of seconds in the VITP.

Figure 5.8 portrays the bandwidth efficiency of our LocVSDPs and the VITP with different return conditions. Figure 5.8(a) illustrates the bandwidth efficiency of the compared protocols when 10 different service providers of the requested service exist in the desired region of interest. Figure 5.8(b) shows the bandwidth efficiency of the compared protocols when there is no service provider that provides the requested service in the specified region of interest. We notice from the curves that our LocVSDPs outperform the VITP with different return conditions in terms of bandwidth efficiency. The bandwidth efficiency of both the naive and the election-based LocVSDP is more than 95 percent for a number of requests ranging from 100 to 1000 requests. However, the bandwidth efficiency of the VITP for 1000 requests is less than 30 percent for a return condition equal to 1 or 3, less than 20 percent for a return condition equal to 5, less than 10 percent for a return condition equal to 7, and is equal to 0 percent for a return condition equal to 10. Thus, we notice that the bandwidth efficiency of the VITP decreases with the increase of the return condition. This is mainly related to its low success rate. In fact, the bandwidth efficiency metric measures the percentage of the bandwidth used for successful transactions. Since the success rate in the VITP decreases with the increase of the return condition, its bandwidth efficiency also decreases with the increase of the return condition.

The bandwidth efficiencies of our LocVSDPs outperform that of the VITP with more than 60 percent. We notice from Figure 5.8(b) that the bandwidth efficiency of the VITP is equal to 0 percent. This is closely related to the success rate as we explained earlier. Thus, if the number of service providers in the desired region of interest is equal to 0, then all the bandwidth used for VITP transactions is unsuccessful and completely inefficient.

5.6 Fault Tolerant EB-LocVSDP

The EB-LocVSDP protocol can work properly assuming that there are no road components and wireless communication link failures. However, this assumption cannot hold all the time in real VANets. For this purpose, we propose in the following fault tolerance mechanisms that permit to the location based service discovery in VANets to succeed in returning a service reply to a requester even in the presence of road component failures and communication link failures.

5.6.1 Description of The Fault Tolerant LocVSDP: FTLocVSDP

In the following, we describe the fault detection and fault tolerance mechanisms that we propose for LocVSDP in order to tolerate the failure of road components and the failure of links. We focus on the crash type of failures.

5.6.1.1 Roadside Router and Link Failures Detection Mechanism

Our fault detection technique is formally described in Algorithm 5.6.1. At each round of the detection mechanism, RRs in the same cluster exchange beacon messages between each others with a time to live (TTL) equal to two. We assume that initially at the protocol bootstrapping, there are no faulty road components and wireless links. Thus, after the first round, each RR learns about its direct RRs neighbors as well as its second RRs neighbors. Every RR has two lists $L1hop$ and $L2hop$ that store its one hop neighbors and its two hops neighbors respectively. Both lists contain neighboring RRs identifiers and the current round number *nRound*. Initially, the round number is equal to one. Then, at the beginning of each new round, RRs increment the round number *nRound* and send again beacon messages. Each RR1 that receives a new beacon message from RR2, checks the TTL and updates the *nRound* field correspondent to RR2 in the following way: if the TTL is equal to one, then RR1 sets the field *nRound* of RR2 in $L1hop$ to the number of the round received in the beacon message. Otherwise, if the TTL is equal to zero, then RR1 sets the field *nRound* in $L2hop$ equal to the round number of the beacon message. At the end of each round, each RR checks the value of the field *nRound* in $L1hop$ for each stored one hop neighbor. If RR finds out that for one of its neighbors RRn the field *nRound* stores a value smaller than the current round number, then it deduces that either RRn is faulty or the link to RRn is faulty. The RR checks the value of the field *nRound* in $L2hop$ for RRn. If $nRound < current(nRound)$ then RR deduces that RRn is faulty. Otherwise, the link between RR and RRn is deduced to be faulty. As a result, our proposed fault detection mechanism permits the detection of faulty RRs and faulty wireless links after each round.

Permanent and intermittent failures can be handled in our proposed FTLocVSDP. In fact, for intermittent failures, and after a RR recovers, it sets the value of the field *nRound* to the current round number received in a beacon message from a neighboring RR. Then, in the next round it sends its own beacon message that will be received by the surrounding RRs. These latters, update their $L1hop$ and $L2hop$ accordingly.

Algorithm 5.6.1 FAILURE DETECTION MECHANISM AT RRI

```
Case O. At the reception of a beacon message from RRj
if Step O1. TTL = 1 then
    Step O1.1. L1hop(j)(nRound) = nRound
end if
if Step O2. TTL = 0 then
    Step O2.1 L2hop(j)(nRound) = nRound
end if
Case P. Periodically:
Step P1. RRi sends a beacon message with TTL = 2
for Step P2. every RRj in L1hop do
    if Step P2.1. L1hop(j)(nRound)<nRound then
        if Step P2.1.1. L2hop(j)(nRound)<nRound then
            Step P2.1.1.1 Neighbors(j).status = faulty
        else
            Step P2.1.2.1 Links(i)(j).status = faulty
        end if
    end if
end for
```

5.6.1.2 Roadside Routers and Links Fault Tolerance Proposed Mechanisms

In our proposed FTLocVSDP, we revisit each phase of the EB-LocVSDP, we discuss its weaknesses with RR and link failures, then we propose our fault tolerance mechanisms.

1. **Fault tolerance during the service request propagation phase:** In the LocVSDP, the location based service request message is propagated towards the region of interest RI. Only RRs and vehicles closer to the center of the RI than the sending road component can forward the request message. Other road components ignore the request message. Thus, if there are failed RRs or failed links between RRs in the forwarding zone to the RI, then a request message could never bypass the failed part of the VANet using the location-based request propagation mechanism of the LocVSDP. That is why, in our FTLocVSDP, we propose a fault tolerant location based request propagation technique, that works efficiently even with RR and link failures. A RR1 can detect the failure of its neighbors or the links to its neighbors through the fault detection mechanism. Assuming that RR1 knows the geographic location of its neighbors, RR1 can know the RRs that are closer to the center of the RI than itself. If RR1 finds out that a RR or a link to a RR closer to the center of the RI than itself is faulty, it sends the location based request message with a ”force broadcast” flag. Neighboring RRs that receive the request message with this flag, rebroadcast the request message whether or not they are

closer to the center of the RI than the sending road component. This way, the request message's forwarding zone is enlarged and the request message can find an alternative path to the RI.

2. **Fault tolerance during the leader election and service reply generation phase:**

   (a) ***Case of leaves failure in the spanning tree:*** In LocVSDP, only the clustered RRs in the RI can participate in the construction of the spanning tree. Assuming that after the construction of the spanning tree, if a leaf RR fails or a link between a leaf RR and its parent RR fails, then all service providers information collected from the leaf RR will not be returned to the leader RR. In the LocVSDP, each RR in the spanning tree waits for a predetermined period of time then it sends its local reply to its parent. However, the leader will not be able to receive all the service providers information located inside the RI. In order to tolerate failures after the construction of the spanning tree and to guarantee that the requester receives information about all the requested service providers inside the RI, we propose a fault tolerance mechanism that is based on the redundancy of service providers information in the spanning tree. Periodically, every RR sends its knowledge about service providers to its neighbors including service providers life time. A service provider information is deleted from a service table if and only if its life time has expired and that it was not updated before its expiration.

   (b) ***Case of parent or link to a parent failure in the spanning tree:*** The failure of a parent RR or a link between two RRs in the spanning tree leads to its partition. Consequently, the child RR will not be able to send its local reply message to its parent that will be propagated to the leader. In our proposed FTLocVSDP, a child node that detects the failure of its parent, or the failure of the link that leads to its parent, chooses another parent RR in the spanning tree. For this purpose it follows the same parent selection process while excluding the detected faulty RRs or the

RR reached through the faulty link. As a result, if the current RR has at least one neighboring RR with a closer distance to the center of the RI, then it stays in the follower state and sends its local reply to the newly selected parent. However, if the current RR has the closest distance to the center of the RI, then it changes to the leader state, it waits for local replies from its children, and finally it aggregates service providers information in a unique reply message that will be propagated to the requester.

(c) ***Case of leader failure in the spanning tree:*** A leader in the spanning tree is also considered as a parent. Thus, if a leader fails before it sends its unique aggregated service reply message and its failure is detected by a child RR. This latter, checks if its distance to the center of the RI is the closest among its neighbors. In this case, it moves to the leader state, and waits for local service replies from its children before it sends its aggregated unique service reply to the service requester. If the current RR finds in its neighborhood another RR that is closer to the center of the RI than itself, then it chooses that RR as its new parent in the spanning tree and sends its local reply.

3. **Case of failure during the service reply propagation phase:** The service reply message is sent to the requester using the CLA-S [40] routing protocol. In this protocol, only road components in a predetermined forwarding zone are supposed to forward the reply message. Thus, in order to tolerate RR and link failures in the forwarding zone, a RR that detects the failure of another RR or a link to another RR that is inside the forwarding zone, sends the service reply message with a ”force broadcast” flag. This way, the service reply message will be propagated to the service requester through alternative paths avoiding the faulty RRs and links.

4. **Case of service provider failure:** A service provider can fail either during the service discovery process or after the discovery process. In the first case, when a service

provider failure is detected by a RR, then this latter does not send the failed provider information to a requester. If the service provider fails after a service reply has been sent to the requester, then the RR that detects the failure of the provider chooses another service provider of the same requested service from its service table and redirects all the traffic coming from the requester to the new selected service provider.

5.6.2 Proof of Correctness of FTLocVSDP

The correctness of FTLocVSDP is proved with the accuracy and correctness of the protocol. Before we discuss FTLocVSDP correctness, let us first present the following lemmas.

Lemma 5.6.1. ***(Service request message propagation)*** *In our proposed FTLocVSDP, the location based service request is propagated in the request zone and it reaches the region of interest RI in a finite time with the presence of faulty RRs and faulty links.*

Proof: In a VANet comprising faulty RRs and links, the LocVSDP protocol fails in some cases to propagate the location based service request message to the desired region of interest RI even if the non faulty RRs are connected to each other through intermediate non faulty RRs or vehicles. In fact, the service request in the LocVSDP is propagated in a predetermined request zone, where only RRs and vehicles closer to the center of the RI should forward it. A RR or a vehicle that is not part of the predetermined request zone does not forward the request message. Our assumption guarantees that non faulty RRs and vehicles are connected to each other but does not guarantee that non faulty RRs and vehicles in a predetermined request zone are connected. Consequently, if non faulty RRs and vehicles in a request zone are not connected, then the LocVSDP would fail to propagate the location based request to the desired RI.

As opposed to the LocVSDP, our proposed FTLocVSDP guarantees the propagation of the service request message to the service provider in a connected VANet with the presence of the failure, even if non faulty RRs and vehicles in a predetermined request zone are not connected. If fact, in FTLocVSDP, a non faulty RR is aware of its neighboring faulty RRs and faulty links through the fault detection mechanism explained in *Cases O and P* in Algorithm 5.6.1. Thus, the current RR is able to determine whether or not the location based request message would be propagated or interrupted due to a faulty RR or a faulty link in the request zone. If the current RR depicts the failure of the next RR or the failure of a link to a RR inside the request zone, then it sends the request message with an indication to force rebroadcast the request message by all the non faulty neighboring RRs. Since we assume that non faulty RRs are connected in the VANet either through intermediate non faulty RRs or through vehicles, our FTLocVSDP guarantees the propagation of the location based request message even through RRs outside the request zone which permits to the request message to reach the RI.

Lemma 5.6.2. ***(Service reply message propagation)*** *In our FTLocVSDP, service reply messages reach their correspondent service requesters in a finite time with the presence of faulty RRs and faulty links.*

Proof: In the LocVSDP, the service reply message is sent from the service provider to the service requester through the CLA-S [40] routing protocol. This latter determines a predetermined forwarding zone of a message to reach its destination. If there are faulty RRs inside the propagation zone between the service requester and the current RR, then the reply message might never reach its destination. In our proposed FTLocVSDP, if a current RR depicts the failure of the next hop RR or a link failure to the next hop RR, then it sends the service reply message with a ”force broadcast” tag. Thus, any RR that receives the reply message with this flag, rebroadcasts the message. Consequently, this guarantees that the service reply message can find alternative routing paths to the destination after enlarging the forwarding zone even if there are RR and link failures in the VANet.

Lemma 5.6.3. ***(Directed spanning tree with presence of failures)*** *Assuming that there are faulty RRs is the VANet, our FTLocVSDP algorithm constructs always a directed spanning in every cluster in the region of interest of a driver request in a finite time. The parent of any Roadside Router RRk in a cluster in the RI is the Roadside Router with the minimal distance to* *the origin of the RI, and having the minimal identifier in case more than one Roadside Router has the minimal distance to the origin of the RI in the same cluster of RRk.*

Proof: Assuming that after a fault-free RR chooses its parent, this latter becomes faulty or the link to this latter becomes faulty, the construction of the spanning tree fails and a deadlock could occur. However, our FTLocVSDP always succeeds in constructing a spanning tree for every service request in a finite time even if a RR or a link to a RR fails after this RR has been chosen as parent or leader. In fact, the failure detection mechanism in our FTLocVSDP provided in *Case O* and *Case P* in Algorithm 5.6.1 permits to each RR to know about its faulty neighboring RRs and links. Thus if the current RR has already chosen its parent and detects the failure of its parent or the link to its parent, then it chooses another fault free parent and sends again its local reply to its parent. Consequently, through FTLocVSDP the process of spanning tree construction would terminate in a finite time even with the presence of faulty RRs and faulty links.

Every RR should select its parent in the eventual spanning tree from its neighboring RRs so that the selected parent is non faulty, and has the shortest distance to the origin of the region of interest. If the current Roadside Router notices the same distance to the origin of the RI for more than one of its neighbors and this distance is the minimal distance in the neighborhood, then the minimal identifier is used to break the ties. Thus, the parent would be the fault-free RR with the minimal distance to the origin of the RI and having the minimal identifier in case more than one Roadside Router has the minimal distance to the origin of the RI in the neighborhood. Roadside routers in the RI are not necessarily connected when RR and link failures are depicted. Thus, a RR can extend the RI in order to be able to select its parent.

Lemma 5.6.4. ***(Cycles avoidance in the constructed spanning tree with presence of failures)*** *Assuming that there are faulty RRs is the VANet, the leader election and spanning tree construction mechanism in our FTLocVSDP protocol ensures the avoidance of deadlocks and terminates in the construction of a directed spanning tree in each cluster inside the RI.*

Proof: We want to prove that there are no cycles in the constructed spanning tree even with the presence of RR and link failures. Let us assume that the number of Roadside Routers in a cluster Ci inside the region of interest RI is n_RRs_FT and the number of links in the same cluster Ci of the region of interest is $n\_links\_ST\_FT$. If we suppose that there is at least one cycle in the constructed spanning tree in Ci, then:

$$n\_links\_ST\_FT > n\_RRs\_FT - 1. \tag{5.17}$$

In FTLocVSDP, every Roadside Router of Ci in the RI selects a unique parent during the election process. If the selected parent RR or the link leading to the selected parent is faulty, then another parent is selected. Besides, the RI could be extended to guarantee the connection between non faulty RRs. The selected parent is the fault free Roadside Router inside Ci with the minimal distance to the origin of the RI and having the minimal identifier in case more than one Roadside Router have a minimal distance to the origin of the RI in the neighborhood. In a cluster Ci comprising RRs, there is at least one Roadside Router that satisfies this property assuming that the fault free RRs are connected to each others inside Ci. Thus, at least one Roadside Router inside Ci will not select a parent. The number of links $n\_links\_ST\_FT$ in the cluster Ci would satisfy the following equation:

$$n\_links\_ST\_FT \leq n\_RRs\_FT - 1 \tag{5.18}$$

This is a contradiction. Therefore, our assumption is false and the lemma is proved.

Lemma 5.6.5. ***(Correct Spanning tree construction with the presence of failures)*** *Assuming that there are faulty RRs is the VANet, our FTLocVSDP protocol ensures the avoidance of deadlocks and terminates the construction of a directed, unique, and connected spanning tree inside every cluster of the region of interest RI of a driver's or passenger's request.*

Proof: Our FTLocVSDP protocol constructs a connected and directed spanning tree even with the presence of RR and link failures in every cluster Ci as established in Lemma 5.6.3. It ensures the avoidance of deadlocks and terminates in the construction of the directed spanning tree in every cluster Ci as established in Lemma 5.6.4.

Finally, since every spanning tree construction process is identified uniquely, the constructed spanning tree in every cluster Ci is unique. Thus, the Lemma 5.6.5 is proved.

Lemma 5.6.6. ***(Leader accuracy and uniqueness with the presence of failures)*** *Assuming that there are faulty RRs is the VANet, the elected leader in every cluster during the resolution of a service request has the shortest distance to the origin of the region of interest RI and is unique.*

Proof: In the following, we prove that the elected leader in a cluster Ci for a driver's or passenger's request is accurate and unique with the existence of faulty RRs and links in an eventually extended RI. It is worth noting that with the presence of RR and link failures the region of interest could be extended in order to guarantee the connectivity between members of the spanning tree. In Lemma 5.6.5, we proved that our FTLocVSDP ensures the avoidance of deadlocks and terminates in the construction of a directed, unique and connected spanning tree related to a requester in every cluster Ci of the RI. The root of the constructed spanning tree has the shortest distance to the origin of the region of interest RI in Ci and is unique. The elected leader Roadside Router related to a driver's request is the root of the constructed spanning tree in the cluster Ci. Thus, the Lemma 5.6.6 is proved and the elected leader inside Ci in the RI during the resolution of a service request has the shortest distance to the origin of the RI and is unique.

Lemma 5.6.7. ***(FTLocVSDP Accuracy)*** *Assuming that there are faulty RRs is the VANet, our FTLocVSDP finds a service that satisfies the requesting vehicle inside the desired region of interest with the presence of faulty Roadside Routers and faulty links in a finite time if the requested service is provided in the region of interest in the VANet.*

Proof: Let us prove the accuracy of our FTLocVDP. With the presence of faulty RRs and faulty links, the LocVSDP might fail to be accurate in finding the requested service because of the disconnections that can occur during the service request propagation or spanning tree construction or service reply propagation phases respectively. However, in FTLocVSDP, we proved in Lemma 5.6.1, that a location based service request gets propagated to the service provider even with the presence of RR and link failures. In Lemma 5.6.5 we proved that the spanning tree is constructed correctly in each cluster inside the RI with the presence of RR and link failure. Finally, in Lemma 5.6.2 we proved that the service reply succeeds to be propagated until the service requester even if there are failures. Thus, the discovery process is not interrupted and assuming that the requested service exists in the region of interest, our FTLocVSDP succeeds in finding all the service providers of the requested service in the desired RI in a finite time.

Lemma 5.6.8. ***(FTLocVSDP Correctness)*** *Our proposed FTLocVSDP will eventually terminate in a finite time with the presence of Roadside Router and link failures.*

Proof: With the presence of faulty RRs and faulty links, the LocVSDP might fail to be correct because of the disconnections that can occur during the service request propagation, the spanning tree construction, or the service reply propagation phases respectively, and because of the deadlocks that can occur as a consequence. However, in FTLocVSDP we proved in Lemma 5.6.1 that a location based service request gets propagated to the service provider even with the presence of RR and link failures. In Lemma 5.6.5 we proved that the spanning tree is constructed correctly with the presence of RR and link failure. Finally in Lemma 5.6.2, we proved that the service reply succeeds to be propagated until the service requester even if there are failures. Thus, the discovery process is not interrupted and our FTLocVSDP terminates in a finite time even with the presence of Roadside Router and link failures.

5.6.3 FTLocVSDP Performance Evaluation

In order to evaluate the performance of our proposed FTLocVSDP, we performed various experiments through the network simulator NS2 [4]. In our simulations, we considered a VANet comprising vehicles and clustered RRs as explained in the system model. The density of the VANet in terms of vehicles per meter square is 120. RRs are clustered mainly around service providers, congested road sections in the VANet, and road sections that suffer from intermittent connections. In our evaluation, we compare the LocVSDP and the FTLocVSDP using the Manhattan mobility model. Details on the emulation of vehicles movements on streets could be found in [35].

In the course of our experiments, we used the IEEE 802.11 as wireless medium with a data transmission rate of 11Mbps and a transmission range of 200 meters. We used the CLA-S [40] as a routing protocol with few modifications. We considered 10 service providers providing the same service and located inside the same region of interest RI. Besides, we assumed 40 service clients distributed along the VANet and requesting the same service located in a region of interest RI of radius 100 meters. Service clients can send many service requests during the simulation time where the arrival time of service requests follows an exponential distribution. We considered that the service request rate of all service clients is the same. We examined a large number of test cases before we provide our results.

In our experiments, we simulated the two types of failures: *RR failures*, and *link failures*. Failures are assumed to be permanent. For RR failures we considered three scenarios: (i) zero faulty RRs, (ii) three faulty RRs located in the forwarding zone of 25 percent of the service clients vehicles, and (iii) seven faulty RRs located in the forwarding zone of 50 percent of the clients vehicles. For link failures, we considered as well three different scenarios: (i) zero faulty links, (ii) three faulty links located in the forwarding zone of 25 percent of the service clients vehicles, and (iii) seven faulty links located in the forwarding zone of 50 percent of the clients vehicles. The main reason of link failures is related to interference problems.

In our experiments, we define the request number as the number of service requests sent by a service client vehicle. For the performance evaluation of the LocVSDP and FTLocVSDP we use three different metrics:

- *Success rate* which indicates the average fraction of successful service transactions over the total number of service requests during the simulation time;

- *Average response time* which calculates the average time of successful request transactions. It measures the elapsed time for getting a valid service reply in response to a service request sent by a vehicle. This metric takes into account several factors such as transmission and message processing delay, just to mention a few.

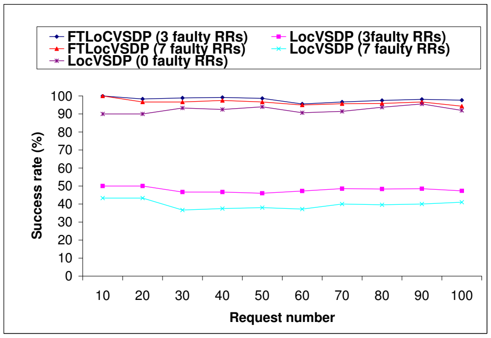
- *Bandwidth usage* which provides the total bandwidth used for the service discovery during the simulation time.

In our experiments, service requests' numbers range from 10 to 100. We generate the initial time of service requests in a random way. We average several runs with a confidence of 95 percent. For every run, we generate randomly a new seed. We summarize the results in graphs 5.9(a), 5.9(b), 5.9(c), 5.10(a), 5.10(b) and 5.10(c). In the following, we discuss our simulation results.

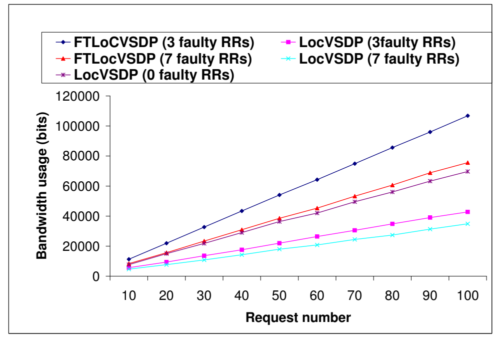
5.6.3.1 Performance Evaluation with Roadside Router Failure

We depict from Figure 5.9(a) that the performance of the LocVSDP is drastically affected with the presence of RR failures. In fact, for three RR failures, the success rate is only 50 percent. It decreases to 40 percent when 7 RR failures are simulated. The low success rate in the LocVSDP is mainly due to the service requests interruption during the location based request propagation phase and to the service reply messages interruption during the reply propagation phase. In the request propagation phase, faulty RRs do not propagate the requests messages. As we assumed earlier, 25 percent of the service clients have the faulty RRs in their forwarding zone. Thus, since client services have the same rate of sending requests, then 25 percent of the total number of service request messages will be interrupted during their propagation. Service queries are also interrupted during the reply propagation phase. In fact, if a service reply message cannot reach the service requester, then the service query will fail.

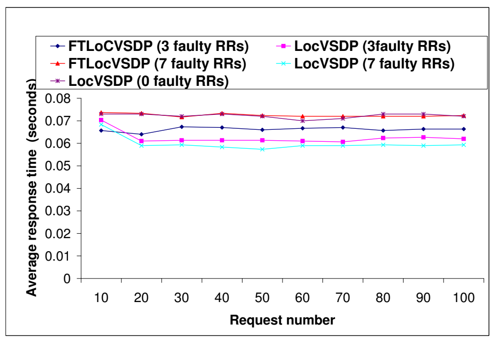
In a similar way, the scenario with 7 faulty RRs has only 40 percent of the service queries satisfied. This is mainly due to the fact that 50 percent of service clients have the faulty RRs



(a) Success Rate comparison



(b) Bandwidth usage comparison



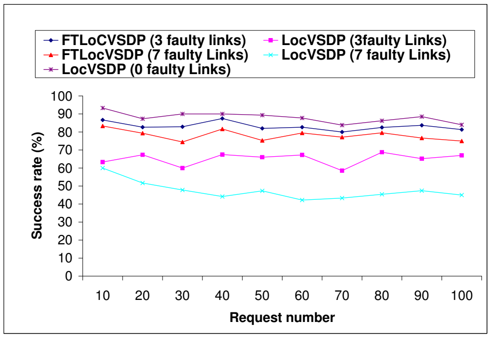
(c) Average response time comparison

Figure 5.9: Performance evaluation of the Fault Tolerant LocVSDP protocol with Roadside Router failures in a City traffic scenario

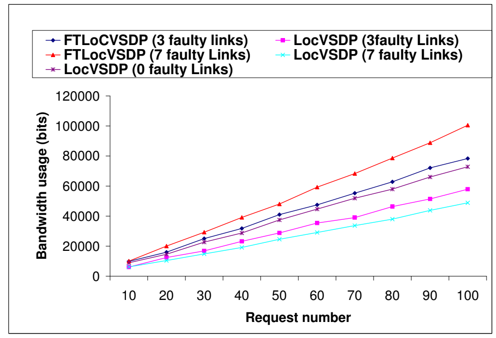
in their forwarding zone of service requests. Consequently, at least 50 percent of the service queries are interrupted during the request propagation phase. The other 10 percent of service queries are interrupted during the service reply propagation phase. Our proposed FTLocVSDP shows better performance than the LocVSDP in terms of success rate in both scenarios: the three faulty RRs scenario and the seven faulty RRs scenario. Figure 5.9(a) shows the good performance of our proposed FTLocVSDP in terms of success rate, which is increased by almost 50 percent compared to the LocVSDP. In fact, in the seven faulty RRs scenario, our FTLocVSDP has almost 95 percent of the service queries satisfied. Whereas, the LocVSDP has only 40 percent of the service queries satisfied when the number of requests varies between 10 and 100. The reason is that, as opposed to the LocVSDP, in the FTLocVSDP service queries are not interrupted during the service request and the service reply propagation phases, due to the presence of faulty RRs. In fact, using our fault detection and fault tolerance techniques, RRs outside the forwarding zone of service request and reply messages forward the discovery messages when they detect the failure of RRs inside the expected forwarding zone. This permits to find alternative ways to a destination. We conclude that our proposed FTLocVSDP is a promising service discovery protocol that can perform efficiently even with the presence of faulty road components. It is shown in Figure 5.9(b), that the bandwidth usage of our proposed FTLocVSDP is a bit larger than that of the LocVSDP. This is expected because in the FTLocVSDP, extra messages are required for the fault detection and the fault tolerance mechanisms. First, during the fault detection mechanism, neighbor discovery messages are exchanged between RRs in order to detect the faulty RRs. Second, the fault tolerance mechanisms during the service request propagation and service reply propagation phases require sending extra request and reply messages respectively in order to find alternative routing paths towards the destination. However, in the LocVSDP service discovery messages propagation is interrupted if the RRs in the forwarding zone are faulty. That is why the bandwidth usage in the FTLocVSDP is larger than that of the LocVSDP.

We depict also from Figure 5.9(b) that the LocVSDP with faulty RRs incurs more overhead than the LocVSDP with zero faulty RRs. The reason is that at least 50 percent of the service requests are not satisfied and are interrupted as discussed in Figure 5.9(a). Besides, we notice that the bandwidth usage of FTLocVSDP with 7 faulty RRs is less than that of the FTLocVSDP with 3 faulty RRs. This is because, when the number of faulty RRs is increased, there are less possibilities to exchange messages between RRs in the VANet.

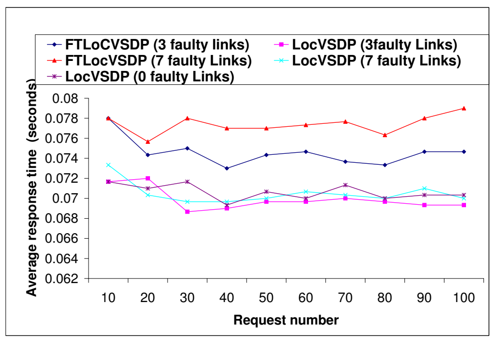
Figure 5.9(c), portrays the curves related to the average response time of the proposed FTLocVSDP and the LocVSDP when 0, 3 and 7 RRs are faulty respectively, and when the number of service requests ranges from 10 to 100. We depict from the Figure 5.9(c) that the average response time of FTLocVSDP with 3 faulty RRs is less than that of the FTLocVSDP



(a) Success Rate comparison



(b) Bandwidth usage comparison



(c) Average response time comparison

Figure 5.10: Performance evaluation of the Fault Tolerant LocVSDP protocol with Link failures in a City traffic scenario

with 7 faulty RRs. The reason is that when the number of faulty RRs is increased, service request and service reply messages are routed through longer paths to their destinations in terms of number of hops, which increases the time required for a message to travel from a source to a destination. Besides, we notice that the average response time of the FTLocVSDP and the LocVSDP without faulty RRs is larger than that of the LocVSDP with faulty RRs. This is because the successful service discovery transactions in the LocVSDP with faulty RRs have service providers and service requesters locations close to each others.

5.6.3.2 Performance Evaluation with Link Failures

Figure 5.10(a) portrays the success rate of FTLocVSDP and LocVSDP when 0, 3 and 7 links between RRs are faulty and when the request number ranges from 10 to 100. The curves related to the FTLocVSDP show better results than the ones related to the LocVSDP when we assumed that 25 percent and 50 percent of service clients have three and seven failed RR links respectively in their forwarding area. In the first scenario where 25 percent of service clients have the three faulty links in their forwarding zone, the success rate of the LocVSDP is in the order of 70 percent. For the second scenario where 50 percent of service clients have the seven faulty links in their forwarding range, the success rate of the LocVSDP is around 50 percent. As for the RR failures scenarios, the LocVSDP has low success rate due to the interruption of service discovery request and reply messages during the request propagation and reply propagation phases respectively when there are faulty links in the forwarding zone.

The FTLocVSDP has a better performance than the LocVSDP as depicted from Figure 5.10(a). In fact, the success rate of FTLocVSDP outperforms the success rate of the LocVSDP by 30 percent in the presence of link failures. This is because, our proposed fault detection mechanism in FTLocVSDP permits the detection of faulty links in the VANet. Then, our proposed fault tolerance mechanism allows the propagation of the discovery messages through alternative links towards their destination, even if those alternative links are outside the expected forwarding zone. Thus, this proves again the good performance of our proposed FTLocVSDP and the benefit of using it in order to increase the success rate of service discovery queries.

In Figure 5.10(b), we present the curves related to the bandwidth usage of FTLocVSDP and the LocVSDP when zero, three, and seven links are faulty respectively. The FTLocVSDP incurred more bandwidth usage than the LocVSDP as we expected. The main reason is that our proposed FTLocVSDP allows the propagation of service discovery messages through links outside the predetermined forwarding zone if the links in the forwarding zone are faulty. We notice from Figure 5.10(b) that the bandwidth usage of FTLocVSDP when 7 links are faulty is greater than that of FTLocVSDP when 3 links are faulty. This is because in our fault tolerance mechanism, we use a ”force broadcast” tag in the service discovery messages when a faulty link is detected. Thus, any RR that will receive the discovery message will rebroadcast it in its range even if it is outside the forwarding zone. Thus, the more link failures are detected, the more RRs are going to rebroadcast the service discovery messages. Figure 5.10(b) shows that the LocVSDP with link failures has a lower bandwidth usage than the LocVSDP without link failures. The main reason is that the success rate of the LocVSDP when there are faulty links is lower than when there are no faulty links. Consequently, service queries are interrupted and less service discovery messages are exchanged, which decreases the bandwidth usage.

Figure 5.10(c) portrays the curves related to the average response time of successful service discovery transactions for the FTLocVSDP and the LocVSDP when zero, three, and seven faulty links are simulated respectively, and when the number of requests ranges from 10 to 100. We notice that the LocVSDP has less average response time than the FTLocVSDP. The main reason is that service discovery messages in the FTLocVSDP take longer routing paths in terms of number of hops from one source to its destination in order to avoid faulty links in the forwarding zone. However, the increase of the average response time in FTLocVSDP is only in the order of milliseconds. Thus, this increase is acceptable and it is not significant as long as the success rate is improved considerably.

5.7 Load-Balancing and QoS-Aware EB-LocVSDP

Load balancing and QoS are very important features for service discovery in VANets. In fact, for VANets that have multiple service providers of the same service, it is very important to balance the load between the different service providers while guaranteeing QoS provisioning, such as satisfying the drivers and passengers queries.

In our service discovery protocol, we design novel techniques for a load balanced and QoS aware service discovery protocol in infrastructure based VANets. First, we guarantee that service requests are handled equitably by service providers in the desired region of interest. For this purpose, it is not necessary that the service provider located closer to the center of the region of interest is returned as a reply to the service requester. However, any service provider located inside the region of interest and that is less loaded than the other service providers in the same region of interest can handle the service request. Second, since our protocol is integrated into a routing protocol, we guarantee that routes between service providers and service requesters are not congested. For this purpose, we propose a novel technique that guarantees load balancing on Roadside Routers in the VANet. In addition to guaranteeing load balancing among the different component of the VANet, our QoS-aware discovery protocol finds service providers and routing paths between service providers and service requesters that satisfy some attributes specified in the user request. Such attributes could be the streaming ratio (for example, in a streaming service, the user specifies the streaming rate required). In the following we describe our proposed techniques for load balancing and QoS aware service discovery protocol in infrastructure based VANets.

5.7.1 Description of The Load Balanced and QoS-Aware LocVSDP

In the following, we describe the load balancing and QoS mechanisms that we propose for the LocVSDP.

1. **Load Balancing on Service Providers:** We define the load on a service provider SP as the number of requests handled by the SP during the simulation time. In our LocVSDP, the service requester receives in the service reply the information of the closest requested service provider to the center of the desired region of interest. If many service requesters specify the same region of interest, then the same service provider SP_0 would be returned to these service requesters, even if there are other service providers located in the desired region of interest. Thus, the service provider SP_0 would be overloaded while the other service providers located in the region of interest are underloaded. This can affect the performance of the overloaded service provider. In order to overcome this weakness

and balance the load on service providers in the VANet, we propose the following load balancing techniques: When a service client sends its request to the desired region of interest, the Roadside Routers inside the region of interest compute a spanning tree with a unique leader RR. Then all RRs send their local replies to the leader of the spanning tree starting from the leaves. In their local replies, every RR adds the load on each service provider in its range. The leader RR receives all the information about SPs including their loads. In the reply returned to the service requester, the leader sends the information of the closest under loaded service provider to the center of the region of interest. Then, the service requester connects to the service provider. This latter updates its load and acknowledges its neighboring Roadside Routers about its new load that would be considered for future requests.

2. **Load Balancing on Roadside Routers Leaders inside the RI:** In our LocVSDP protocol, a leader is elected in the region of interest specified in a user query. The elected leader RR is the closest RR to the center of the region of interest. If many requesters specify the same region of interest, then the load on the same elected leader for many requests would be high. In order to balance the load on RRs in the region of interest, we propose the following mechanisms. A Roadside Router RR_0 that participates in an election process and that is elected as leader stores the request number and the information regarding the region of interest *RI*, i.e. the center and the range. Besides, it computes its distance to the center of the region of interest *DistanceRI*. It stores this value in the variable *realDistanceRI* and initiates a variable *computedDistanceRI* with the value of *DistanceRI*. If RR_0 participates in another election process for the same region of interest *RI* it updates the value of *computedDistanceRI* so that *computedDistanceRI* = *computedDistanceRI* + *range* where *computedDistanceRI* is the previously calculated *computedDistanceRI*, and *range* is the range of the region of interest and it is a fixed value. RR_0 sends *computedDistanceRI* in the message *election_IDReq_msg*. Thus, RR_0 will not be elected again as leader for a new service request in the same region of in-

terest. Thus, the second closest Roadside Router to the center of the region of interest would be elected as leader.

Similarly, the new elected leader executes our mechanism so that it will not be elected as leader for the next similar user query. Since *computedDistanceRI* is calculated by adding a fixed value to the real *DistanceRI*, our process guarantees that all Roadside Routers located in the same region of interest are elected as leaders at least once before anyone of them is elected again as leader for the second time. As a result, our technique guarantees load balancing on Roadside Routers in any region of interest requested by many service clients.

3. **Load Balancing in the Routing Path Between Service Providers and Service Requesters:** In our LocVSDP protocol, the routing path established between a service requester and its correspondent service provider is the fastest path, i.e. the path from which the elected leader has received the first request message or parent message. In our QoS and load balancing LocVSDP, the fastest path is not necessarily chosen. If the fastest path is congested, our load balancing and QoS based protocol is able to generate an alternative path less congested than the fastest path in order to establish an efficient communication between the service provider and the service requester. In the following, we explain how our load balancing and QoS based LocVSDP protocol generates the less congested path while propagating the service reply. We define the cost of a routing path as the sum of loads on each component of the routing path. In order to generate the less congested path between a service provider and a service requester, we implement the following technique. After a service request is propagated to the desired region of interest and handled by the elected leader, a service provider is chosen and a unique reply is generated. This latter is geocasted to the service requester from the elected Roadside Router leader. An intermediate *RR* forwards the received reply message only if: *(i)* the *RR* is closer to the service requester than the RR from which it has received the reply message, and *(ii)* the cumulated cost of the routing path of a request *ReqID*

until the current RR is less than a previously cumulated cost of a routing path for the same request *ReqID*. In this way, it is guaranteed that the reply message is not flooded in the VANet and that the routing path returned to the service requester is the less congested one among all the possible routing paths between the service provider and the service requester. If the current intermediate RR has to propagate the reply message, it adds to the reply message its ID, its capacity, its load, and the link load between itself and the previous RR. The service requester sends a connection message to the service provider through the less congested routing path including its eventual load. Thus, each intermediate component updates its load accordingly.

4. **Service Provider Quality Requirement:** Many service providers providing the same service could have different characteristics and quality (such as streaming rate). Service requesters could specify in their requests the quality of the service that they require using different attributes. In our proposed QoS based service discovery technique, the quality attributes specified by a service request are handled at the elected Roadside Router leader. After the reception of all the local service replies from Roadside Routers inside the region of interest RI, the leader Roadside Router can determine the appropriate service provider that meets all the quality attributes defined by the service requester while maintaining the load balancing on service providers.

5. **Intermediate Components Quality Requirement:** Intermediate components (RRs and links) between a service provider and a service requester can have different characteristics and quality. A service requester could require a desired quality in the routing path to a service provider. In our QoS based LocVSDP, we handle the requester requirement during the reply propagation and routing path generation phase. Only routing paths that satisfy the QoS requirements are considered by the service requester.

Table 5.3: Simulation parameters

Parameter name	Parameter value
Wireless medium	IEEE 802.11
Data transmission rate	11Mbps
Transmission range (meters)	200
Average vehicle's Speed (meter/second)	20
Simulation time(seconds)	1200
Simulation area ($meter^2$)	$600 \times 600 = 120,000$
number of vehicles	130
Roadside router number	16
Clients' number	40
servers' number	9
Region of interest surface ($meter^2$)	$\Pi \times 100^2$

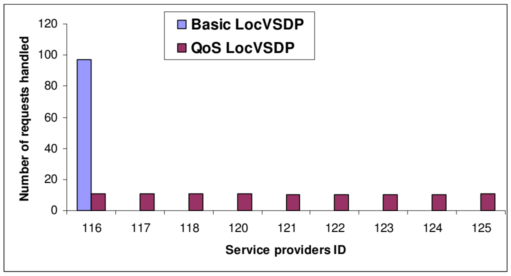
5.7.2 Performance Evaluation of the QoSLocVSDP

In this section, we present the simulation we have conducted to evaluate the performance of our load balancing and QoS-aware LocVSDP in infrastructure based VANets. We performed our simulations on the network simulator NS2 [4]. In our experiments, we simulated an infrastructure-based VANet comprising 16 Roadside Routers. We set the average number of vehicles circulating along the network road sections to 130 vehicles. We used realistic mobility traces [35] in order to evaluate the original, and the load balancing and QoS aware LocVSDPs' performance. As illustrated in our simulation parameters in Table 5.3, we use the IEEE 802.11 as wireless medium with a data transmission rate of 11Mbps and a transmission range of 200 meters. We use the CLA-S [40] as a routing protocol to route communication packets between Roadside Routers and vehicles. We consider 9 service providers located inside the RI and providing the same service requested by the 40 service clients circulating in the VANet. The arrival time of service requests follows an exponential distribution.

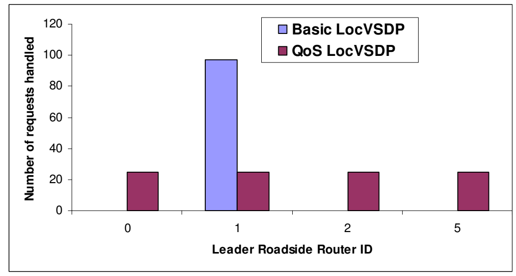
The evaluation of our load balancing and QoS aware discovery protocol required the investigation of various test cases. In our chosen test cases, we assume that service clients' queries target the same service type in the same region of interest. Besides, we assume that the 9 service providers are located in the same region of interest specified by the service clients and that they provide the same service type. Service clients can generate one or many service requests during the simulation time. We assume that 50% of the service requests specify some performance attributes on service providers and/or routing paths. In the course of our experiments, we define the request number as the number of service queries sent by vehicle clients. We use the following metrics for the performance evaluation of the original LocVSDP and the load balancing and QoS based LocVSDP.

- *Success rate* which indicates the average fraction of successful service transactions;
- *Connection rate* which determines the average percentage of successful service connections; i.e. the service requester is able to connect to the service provider through the returned routing path;
- *Average response time* which calculates the average time of successful request transactions. It measures the elapsed time for getting a valid service reply in response to a service request sent by a vehicle. This metric takes into account several factors such as transmission and message processing delay, just to mention a few;
- *Bandwidth usage* which measures the bandwidth usage of driver's service queries during the simulation time;
- *Average load* which measures the average load on vehicular components such as service providers and roadside routers.

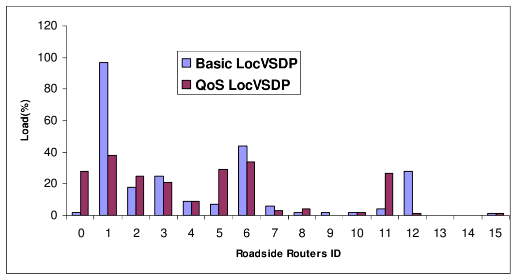
The results of our extensive simulation experiments are illustrated in Figures 5.11 and 5.12. They are obtained after we averaged several runs with a value of confidence equal to 95 percent. Figure 5.11(a) illustrates the average load in terms of average number of requests handled by each service provider located in the region of interest *RI*. In this variant, we set the number of requests to 100. In the LocVSDP protocol 97% of the requests are handled by the service provider 116, whereas in the QoS based LocVSDP every service provider located in the RI handles at most 11% of the total requests generated during the simulation time. This proves that our QoS LocVSDP succeeds in balancing the load among the different service providers in the RI. In the LocVSDP protocol, the service reply contains information about



(a) Comparison of load balancing on Roadside Routers inside the region of interest of the original LocVSDP and the QoS LocVSDP protocols using a city mobility model



(b) Comparison of load balancing on service providers of the original LocVSDP and the QoS LocVSDP protocols using a city mobility model

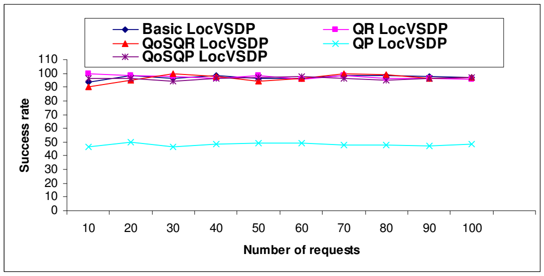


(c) Comparison of load balancing on Roadside Routers in the geocast region of the VANet of the original LocVSDP and the QoS LocVSDP protocols using a city mobility model

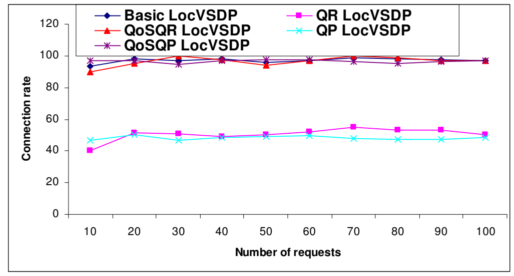
Figure 5.11: Performance evaluation of the load balanced LocVSDP

the closest service provider to the center of the region of interest. Since in our simulation scenario we assume that all the service requests specify the same region of interest, the closest service provider 116 located closer to the center of the region of interest than the other service providers handled 97% of the total requests and 3% of the total requests were dropped. However, in the QoS LocVSDP, service replies contain information about the closest less loaded service provider to the center of the RI. Thus, 99% of the total number of requests are handled by service providers inside the RI in such a way that every service provider handles at most 11% of the total requests.

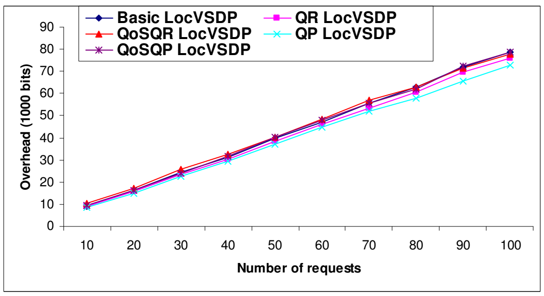
Figure 5.11(b) depicts the load balancing on Roadside Routers inside the region of interest *RI* of the LocVSDP and the QoS LocVSDP respectively for 100 requests. Four Roadside Routers are located inside the region of interest *RI*, their IDs are 0, 1, 2 and 5 respectively. The



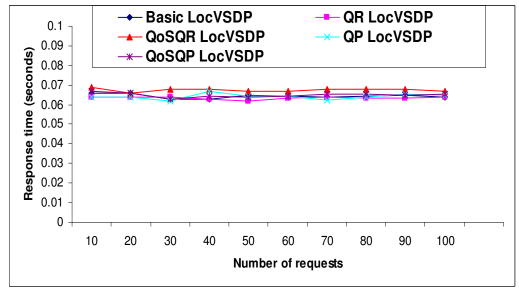
(a) Success rate comparison



(b) Connection rate comparison



(c) Bandwidth usage comparison



(d) Average response time comparison

Figure 5.12: Comparison of multiple variants related to the original LocVSDP and the QoS LocVSDP protocols using a city mobility model

Figure 5.11(b) shows that for the LocVSDP, almost 100 percent of the requests are handled by the elected leader Roadside Router having the ID 1, whereas for the QoS LocVSDP, requests are handled equitably by all the Roadside Routers inside the *RI*. In fact, every Roadside Router inside the RI is elected as leader to process requests of at most 25% of the total requests. Thus, our proposed load balancing mechanism succeeds in balancing the processing load of service requests by Roadside Routers inside the *RI*.

Figure 5.11(c) portrays the load balancing on Roadside Routers inside the VANet using the LocVSDP and the QoS LocVSDP respectively for 100 service requests. It shows mainly the used capacity of each Roadside Router in the simulated VANet. In the LocVSDP, almost 100% of the capacity of Roadside Router 1 is used. The used capacity of the other Roadside Routers is less that 50%. The reason behind this is that in the LocVSDP, the service reply returned to the service requester contains information about the fastest path to the service provider, i.e. the path from which the elected leader received the first request message or parent message as we explained earlier. Thus, the LocVSDP, does not consider the load and the capacity of

Roadside Routers in the choice of the routing path between the service provider and the service requester. However, in the QoS LocVSDP, the used capacity of all the Roadside Routers is less than 40%. Thus, our proposed load balancing mechanism succeeds again in guaranteeing a balanced load on the Roadside Routers. In fact, in the QoS LocVSDP, a service reply is geocasted to the service requester and the path retained for the communication between the service provider and the service requester is not necessarily the fastest path, but it is the less loaded path among the traversed routing paths in the geocast region between the service provider and the service requester. This explains the balanced capacity used in Roadside Routers RR0, RR1, RR2, RR3, RR4, RR5, RR6, RR7, and RR11 located inside the geocast region.

In Figure 5.12, we present the curves related to the success rate, the connection rate, the bandwidth usage, and the average response time respectively, of variants of the original LocVSDP and the QoS LocVSDP when the number of requests varies between 10 and 100. The different variants are described below:

- *Original LocVSDP:* corresponds to the execution of the LocVSDP protocol when the service requester does not specify any QoS requirement in its request.

- *QR LocVSDP:* corresponds to the execution of the LocVSDP when the service requester specifies a QoS requirement in the routing path used for communication to the service provider. Some Roadside Routers in the VANet between service requesters and service providers do not satisfy the requested QoS requirement.

- *QoSQR LocVSDP:* corresponds to the execution of the QoS LocVSDP when the service requester specifies a QoS requirement in the routing path used for communication to the service provider. Some Roadside Routers in the VANet between service requesters and service providers do not satisfy the requested QoS requirement.

- *QP LocVSDP:* corresponds to the execution of the LocVSDP when 50% of the total number of requests have a QoS requirement in the requested service provider; and the service provider that have the ID 116 does not satisfy the QoS requirements.

- *QoSQP LocVSDP:* corresponds to the execution of the QoS LocVSDP when the service requester specifies a QoS requirement in the requested service provider.

In Figure 5.12(a), we depict that the success rate of the different variants is quite high (more than 90%) except for the *QP LocVSDP* variant. This latter has a success rate in the order of 50%. *QP LocVSDP* corresponds to the execution of the LocVSDP when 50% of the total number of requests have QoS requirements in the requested service provider. Since the LocVSDP does not take into consideration any QoS requirement and that the closest service provider to the origin of the *RI* is the provider with ID 116, almost 50% of the service requests in this variant cannot be satisfied. This explains the low success rate in the *QP LocVSDP* variant. However, for the *QoSQP LocVSDP*, which corresponds to the execution of the QoS LocVSDP when 50% of the total number of requests have QoS requirements in the requested service provider, has high success rate. This is because our proposed QoS LocVSDP mechanism takes into account the QoS requirements specified in the clients requests. The success rate in the other variants (*QR LocVSDP* and *QoSQR LocVSDP*) is not affected when the requester specifies a QoS requirement in the requested service provider.

Figure 5.12(b) illustrates the curves related to the connection rate of service requesters to their correspondent service providers. We notice that the connection rate of the curve that corresponds to the variant *QR LocVSDP* is less than 50%. This is mainly due to the fact that the LocVSDP returns a service reply to the service requester containing the fastest path to the service provider and does not consider any QoS requirements in the routing path. The fastest path could not have all its routing components satisfy the QoS requirements specified in the client request. Consequently, even if the service provider is found and the service reply is returned to the service requester, which explains the high success rate of *QR LocVSDP* in Figure 5.12(a), this latter cannot establish a communication connection through the returned routing path. That is why the connection rate in the *QR LocVSDP* variant is not high. However, the connection rate in the *QoSQR LocVSDP* variant is high because this latter takes into account the QoS requirements specified in a service request. The curve corresponding to *QP LocVSDP* variant has its connection rate as good as its success rate. This is only because the requesters that receive service replies with the appropriate service provider that can establish a connection with their correspondent service providers.

Figure 5.12(c) portrays the curves of the bandwidth usage of the different variants related to the LocVSDP and the QoS LocVSDP when the number of requests ranges between 10 and 100. We notice that all the curves have the same behavior. The bandwidth usage of the different variants is in the order of 10000 bits for 10 requests and increases to reach around 80000 bits for 100 requests. If we examine more closely the curves, we notice that the bandwidth usage in the variants related to the QoS LocVSDP is slightly more than the bandwidth usage in the variants related to the LocVSDP. This is mainly due to the fact that in the QoS LocVSDP, more messages are exchanged between vehicular components that are related to load balancing and QoS requirements satisfaction purposes. Figure 5.12(d) depicts the average response time of the different variants related to the execution of the LocVSDP and the QoS LocVSDP when the number of requests ranges between 10 and 100. We notice that the average response time of the different variants is in the order of 65 milliseconds. The QoS LocVSDP protocol does not have an impact on the average response time.

We conclude that our proposed load balancing and QoS LocVSDP protocol succeeds in balancing the load between the different components of the VANets (Roadside Routers and service providers) and satisfies clients requests with different QoS requirements, while guaranteeing a good response time and appropriate bandwidth usage.

5.8 Summary

In this chapter, we proposed efficient and scalable location-based service discovery protocols for VANets (LocVSDP, FTLocVSDP, and QoSLocVSDP). A comparison study of our LocVSDP with the existing location-based service discovery protocol VITP has shown that our techniques outperform greatly the VITP in terms of success rate, average response time, bandwidth usage, and bandwidth efficiency. Moreover, the experimental results of the FT-LocVSDP proved that our techniques have a good performance mainly in the success rate of discovering queries while dealing with Roadside Router and link failures. Finally, our proposed QoSLocVSDP succeeded in balancing the load between the different components of the VANet, including Roadside Routers and service providers. In addition, it showed high performances, in terms of response time and bandwidth usage while satisfying clients requests with different QoS requirements.

Chapter 6

Conclusion

The research on vehicular networks has been growing rapidly in recent years. In fact, due to the increasing number of applications and services in vehicular networks, many researchers have been focusing upon the problem of service discovery. In the previously proposed protocols on service discovery, only few of them considered the scalability, the QoS, and the fault tolerant aspects while finding a service provider to a service requester. The lack of doing such improvements, can have a dramatic effect on the performance of vehicular networks since many service queries will be dissatisfied.

6.1 Summary of Contributions

In this thesis we focused on the problem of scalable service discovery protocols in vehicular networks. First, we presented a classification of service discovery protocols based upon their characteristics. Then, we presented our contributions and reported the performance of our algorithms using an extensive set of simulation experiments. The contributions of this thesis are as follow:

- *A scalable infrastructure-less gateway discovery protocol for vehicular networks:*

  We proposed a scalable hybrid adaptive Location-Aided Gateway Advertisement and Discovery protocol (LAGAD). Our protocol combines both reactive and proactive approaches, and determines efficiently the gateway advertisement zone through a location-aided technique. Our results indicated that our LAGAD scheme is scalable and that a significant success rate could be achieved using our algorithm, while guaranteeing low response time (in the order of milliseconds) and low bandwidth usage when compared to other gateway discovery approaches. Moreover, our results indicated that LAGAD achieves a high delivery ratio of data packets, a low end to end delay, and permits duplicate and ordered data packets reception at the destination gateway.

- *A scalable service discovery protocol for vehicular networks with infrastructure support:* We proposed a bandwidth-efficient and scalable hybrid adaptive service discovery protocol (VSDP) for Vehicular Networks that relied on a wireless Roadside Routers infrastructure. Our results indicated that our techniques can achieve significant success rate (more than 90 percent), while guaranteeing low response time (in the order of milliseconds) and low bandwidth usage when compared to existing service discovery techniques.

- *A scalable location-based service discovery protocol for vehicular network relying on a cluster-based infrastructure:* We proposed a new context-aware and location-based service discovery protocol (LocVSDP) for Vehicular Networks that relies on Roadside Routers cluster-based infrastructure and offers a scalable framework for the discovery of time-sensitive and location based services in Vehicular Networks. We discussed the implementation of our protocol and techniques, reported on performance evaluation experiments, and compared against an existing location-based discovery protocol (VITP). Our simulation results indicate that our techniques outperform the VITP protocol in terms of success rate, average response time, bandwidth usage and bandwidth efficiency. In essence, the proposed LocVSDP shows a gain of 20 percent in terms of success rate. LocVSDP uses at least 90 percent less bandwidth than VITP, and its average response time is at least 10 percent lower than VITP for successful query transactions.

- We enhanced our proposed location based service discovery protocol (LocVSDP) such that it can work well under *service providers failures*, *Roadside Routers failures* and *communication links failures*.

- We enhanced further our proposed location based service discovery protocol such that it balances the traffic load between road components and guarantees the satisfaction of drivers' QoS requirements.

6.2 Future Work

We can identify several directions for further research:

- We plan to investigate ideas to support future generation of Vehicular Ad hoc Networks (VANets) that will integrate different wireless network technologies such as IEEE 802.11, 3G, WiMax, among others. This integration will not only provide drivers and passengers with seamless wireless connectivity, but it will also promote a plethora of new applications.

- The security issue is very important for VANets. There are several directions that could be investigated in order to make our protocols secure. We plan to investigate security aspect for our proposed service discovery protocols that will protect service providers and clients from security attacks. We will look mainly into the following aspects: authentication and key management, privacy, and secure positioning.

- In this thesis, we studied the performance of our protocols through simulations. We plan to implement our proposed service discovery protocols in real testbed in order to assess their performance in real scenarios. Such testbed experiments will allow us to enhance our protocols.

- We plan to use our proposed protocols and built a Service Oriented Architecture (SOA) that permits access technologies heterogeneity and services interoperability.

References

[1] http://www.car-accidents.com/pages/stats.html.

[2] http://www.fleetnet.de/.

[3] NAVSTAR GPS operations. *available via WWW at URL: http://tycho.usno.navy.mil/gpsinfo.html*.

[4] The Network Simulator 2. *http://www.isi.edu/nsnam/ns/*.

[5] Urban Sensing Project. In *research.cens.ucla.edu/projects/2006/Systems/Urban_Sensing*.

[6] H. S. AbdelSalam, S. Olariu, and S. R. Rizvi. Tiling-based localization scheme for sensor networks using a single beacon. In *GLOBECOM*, pages 790–794. IEEE, 2008.

[7] A. Adya, P. Bahl, J. Padhye, A. Wolman, and L. Zhou. A multi-radio unification protocol for IEEE 802.11 wireless networks. In *IEEE Broadnets*, pages 344–354, 2004.

[8] P. Bahl, A. Adya, J. Padhye, and A. Walman. Reconsidering wireless systems with multiple radios. *ACM SIGCOMM Computer Communication Review*, 34(5):39–46, 2004.

[9] S. Basagni, I. Chlamtac, V. Syrotiuk, and B. Woodward. A distance routing effect algorithm for mobility (DREAM). *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, pages 76–84, 1998.

[10] P. Basu and T. Little. Wireless ad hoc discovery of parking spaces. In *Workshop on Applications of Mobile Embedded Systems*, 2004.

[11] M. Bechler, L. Wolf, O. Storz, and W. Franz. Efficient discovery of Internet gateways in future vehicular communication systems. In *Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual*, volume 2, 2003.

[12] C. Bettstetter and J. Eberspacher. Hop distances in homogeneous ad hoc networks. In *Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual*, volume 4, 2003.

[13] S. Biswas, R. Tatchikou, and F. Dion. Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety. *Communications Magazine, IEEE*, 44(1):74–82, 2006.

[14] A. Boukerche. Handbook of Algorithms for Wireless Networking and Mobile Computing. *Chapman and Hall/CRC*, 2005.

[15] A. Boukerche. Algorithms and Protocols for Wireless, Mobile Ad Hoc Networks. *John Wiley and Sons Inc*, 2008.

[16] A. Boukerche. Algorithms and Protocols for Wireless Sensor Networks. *Wiley-IEEE Press*, page 544 pages, 2008.

[17] A. Boukerche, H. Oliveira, E. Nakamura, and A. Loureiro. Vehicular ad hoc networks: A new challenge for localization-based systems. *Computer Communications*, 31(12):2838–2849, 2008.

[18] M. Caliskan, D. Graupner, and M. Mauve. Decentralized discovery of free parking places. In *Proceedings of the 3rd international workshop on Vehicular ad hoc networks*, pages 30–39. ACM New York, NY, USA, 2006.

[19] T. Camacho, V. Kostakos, and C. Mantero. A Wireless Infrastructure for Delivering Contextual Services and Studying Transport Behavior. *13th International IEEE Conference on Intelligent Transportation Systems*, pages 943–948, 2010.

[20] T. Camp and Y. Liu. An adaptive mesh-based protocol for geocast routing. *Journal of Parallel and Distributed Computing*, 63(2):196–213, 2003.

[21] A. Campbell, S. Eisenman, N. Lane, E. Miluzzo, and R. Peterson. People-centric urban sensing. In *Proceedings of the 2nd annual international workshop on Wireless internet*, page 18. ACM Press New York, NY, USA, 2006.

[22] C. Campo, M. Munoz, J. Perea, A. Mann, and C. Garcia-Rubio. PDP and GSDL: a new service discovery middleware to support spontaneous interactions in pervasive systems. *Third IEEE International Conference on Pervasive Computing and Communications Workshops, 2005. PerCom 2005 Workshops.*, pages 178–182, 2005.

[23] D. Chakraborty, A. Joshi, Y. Yesha, and T. Finin. GSD: a novel group-based service discovery protocol for MANETS. *4th International Workshop on Mobile and Wireless Communications Network, 2002.*, pages 140–144, 2002.

[24] L. Cheng and I. Marsic. Service discovery and invocation for mobile ad hoc networked appliances. *Proc. 2nd International Workshop on Networked Appliances*, 2000.

[25] M. Dikaiakos et al. Location-Aware Services over Vehicular Ad-Hoc Networks using Car-to-Car Communication. *IEEE Journal on Selected Areas in Communications*, 25(8):1590–1602, 2007.

[26] S. Dolev, S. Gilbert, N. Lynch, E. Schiller, A. Shvartsman, and J. Welch. Virtual Mobile Nodes for Mobile Ad Hoc Networks. *Lecture notes in computer science*, pages 230–244, 2004.

[27] G. Dommety and R. Jain. Potential Networking Applications of Global Positioning Systems (GPS). *Arxiv preprint cs.NI/9809079*, 1998.

[28] Z. Fan and E. Ho. Service Discovery in Mobile Ad Hoc Networks. *Proceedings of the Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM'05)-Volume 01*, pages 457–459, 2005.

[29] R. Fracchia and M. Meo. Analysis and Design of Warning Delivery Service in Intervehicular Networks. *IEEE Transactions on Mobile Computing*, 7(7):832–845, 2008.

[30] C. Gan and Y. Lin. Push-to-talk service for intelligent transportation systems. *IEEE Transactions on Intelligent Transportation Systems*, 8(3):391–399, 2007.

[31] M. Ghassemian, P. Hofmann, C. Prehofer, V. Friderikos, and H. Aghvami. Performance analysis of Internet gateway discovery protocols in ad hoc networks. *IEEE Wireless Communications and Networking Conference, WCNC*, 1, 2004.

[32] P. Gupta and P. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, 2000.

[33] E. Guttman, C. Perkins, J. Veizades, and M. Day. Service Location Protocol, Version 2. Technical report, IETF RFC-2165, November 1998, 1999.

[34] Z. Haas, M. Pearlman, et al. The Zone Routing Protocol (ZRP) for Ad Hoc Networks. *INTERNET DRAFT-Mobile Ad hoc Networking (MANET) Working Group of the bternet Engineering Task Force (ETF), November*, 1997.

[35] J. Haerri, F. Filali, and C. Bonnet. Mobility Models for Vehicular Ad Hoc Networks: A Survey and Taxonomy. *research rep. RR-06-168, Institut Eurecom, Mar*, 2007.

[36] R. Handorean, R. Sen, G. Hackmann, and G. Roman. Context aware session management for services in ad hoc networks. In *IEEE International Conference on Services Computing*, pages 113–120, 2005.

[37] H. Hassanein, Y. Yang, and A. Mawji. A new approach to service discovery in wireless mobile ad hoc networks. *International Journal of Sensor Networks*, 2(1):135–145, 2007.

[38] H. Hassanein, Y. Yang, and A. Mawji. A new approach to service discovery in wireless mobile ad hoc networks. *International Journal of Sensor Networks*, 2(1):135–145, 2007.

[39] S. Helal, N. Desai, and V. Verma. Konark-a service discovery and delivery protocol for ad-hoc networks. *Proceeding of the Third IEEE Conference on Wireless Communications and Networking, WCNC.*, 3, 2003.

[40] A. Ho, Y. Ho, and K. Hua. A connectionless approach to mobile ad hoc networks in street environments. In *Proceedings IEEE Intelligent Vehicles Symposium*, pages 575–582, 2005.

[41] B. Hull et al. CarTel: A Distributed Mobile Sensor Computing System. In *4th ACM SenSys*, pages 125–138, Boulder, CO, 2006.

[42] C. Jelger, T. Noel, and A. Frey. Gateway and address autoconfiguration for IPv6 ad-hoc networks. *IETF draft expired, draft-jelger-manet-gateway-autoconf-v6-02. txt, Apr*, 2004.

[43] D. Johnson and D. Maltz. Dynamic source routing in ad hoc wireless networks. *Mobile Computing*, 353:153–181, 1996.

[44] C. Julien and G. Roman. Active coordination in ad hoc networks. *Lecture notes in computer science*, pages 199–215, 2004.

[45] N. Klimin, W. Enkelmann, H. Karl, and A. Wolisz. A Hybrid Approach for Location-based Service Discovery in Vehicular Ad Hoc Networks. In *in Proc. of 1st Intl. Workshop on Intelligent Transportation (WIT)*. Citeseer, 2004.

[46] Y. Ko and N. Vaidya. Location-Aided Routing (LAR) in mobile ad hoc networks. *Wireless Networks*, 6(4):307–321, 2000.

[47] U. Kozat and L. Tassiulas. Network layer support for service discovery in mobile ad hoc networks. *Proceeding of IEEE INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE*, 3.

[48] T. Kunz. Multicasting in mobile ad-hoc networks: achieving high packet delivery ratios. In *CASCON '03: Proceedings of the 2003 conference of the Centre for Advanced Studies on Collaborative research*, pages 156–170. IBM Press, 2003.

[49] S. Lee, M. Gerla, and C. Chiang. On-demand multicast routing protocol. *IEEE Wireless Communications and Networking Conference WCNC*, pages 1298–1302, 1999.

[50] S.-H. Lee and Y.-B. Ko. A Robust Service Discovery Approach for Hybrid Ad Hoc Networks. *The International Conference on Dependable Systems and Networks (DSN'05)*, 2005.

[51] L. Li and L. Lamont. A Lightweight Service Discovery Mechanism for Mobile Ad Hoc Pervasive Environment Using Cross-Layer Design. *Proceedings of the 3rd IEEE International Conference on Pervasive Computing and Communications Workshops, Kauai Island, Hawaii, March*, 2005.

[52] C. Lochert, M. Caliskan, B. Scheuermann, A. Barthels, A. Cervantes, and M. Mauve. Multiple Simulator Interlinking Environment for Inter Vehicle Communication. In *Proc. 2nd ACM Int. Workshop on Vehicular Ad Hoc Networks (VANET), Cologne, Germany*, pages 87–88, 2005.

[53] W. Lou and J. Wu. On reducing broadcast redundancy in ad hoc wireless networks. *IEEE Transactions on Mobile Computing*, pages 111–122, 2002.

[54] F. Martinez, C. Toh, J. Cano, C. Calafate, and P. Manzoni. Emergency Services in Future Intelligent Transportation Systems Based on Vehicular Communication Networks. *Intelligent Transportation Systems Magazine*, 2(2):6–20, 2010.

[55] A. D. May. Traffic Flow Fundamentals. In *Prentice Hall*, 1990.

[56] C. Merlin and W. Heinzelman. Highway Traffic Generator in C++. *Wireless Communications and Networking Group. www.ece.rochester.edu/research/wcng/code.html.*

[57] C. Merlin and W. Heinzelman. A study of safety applications in vehicular networks. In *IEEE International Conference on Mobile Adhoc and Sensor Systems*, pages 102–109, 2005.

[58] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode. TrafficView: traffic data dissemination using car-to-car communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 8(3):6–19, 2004.

[59] K. Nagel and M. Schreckenberg. A cellular automaton model for freeway traffic. *J. Phys. I France*, 2(2221):222–235, 1992.

[60] K. Nagel, D. Wolf, P. Wagner, and P. Simon. Two-lane traffic rules for cellular automata: A systematic approach. *Physical Review E*, 58(2):1425–1437, 1998.

[61] M. Nidd. Service discovery in DEAPspace. *IEEE Wireless Communications*, 8(4):39–45, 2001.

[62] V. Park and M. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. In *INFOCOM'97. Proceedings IEEE, Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 1405–1413. IEEE, 1997.

[63] B. Parkinson and S. Gilbert. NAVSTAR: Global positioning system Ten years later. *Proceedings of the IEEE*, 71(10):1177–1186, 1983.

[64] C. Perkins, E. Belding-Royer, and S. Das. Ad hoc On-Demand Distance Vector (AODV) Routing. *INTERNET DRAFT, IETF MANET Working Group*, 2003.

[65] C. Perkins and P. Bhagwat. Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers. *Proceedings of the conference on Communications architectures, protocols and applications*, pages 234–244, 1994.

[66] U. Plug. Play Forum, Understanding Universal Plug and Play (UPnP). *White Paper*, 2000.

[67] V. Ramasubramanian, Z. Haas, and E. Sirer. SHARP: a hybrid adaptive routing protocol for mobile ad hoc networks. *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 303–314, 2003.

[68] P. Ratanchandani and R. Kravets. A hybrid approach to Internet connectivity for mobile ad hoc networks. *Wireless Communications and Networking (WCNC)*, 3, 2003.

[69] O. Riva, T. Nadeem, C. Borcea, and L. Iftode. Context-Aware Migratory Services in Ad Hoc Networks. *IEEE Transactions on Mobile Computing*, pages 1313–1328, 2007.

[70] P. Ruiz and A. Gomez-Skarmeta. Adaptive gateway discovery mechanisms to enhance internet connectivity for mobile ad hoc networks. *Ad Hoc and Sensor Wireless Networks*, 1(1):159–177, 2005.

[71] F. Sailhan and V. Issarny. Scalable Service Discovery for MANET. *Proc. of the 3rd IEEE International Conference on Pervasive Computing and Communications (PERCOM)*, pages 235–244, 2005.

[72] P. Samar, M. Pearlman, and Z. Haas. Independent zone routing: an adaptive hybrid routing framework for ad hoc wireless networks. *IEEE/ACM Transactions on Networking (TON)*, 12(4):595–608, 2004.

[73] G. Schiele, C. Becker, and K. Rothermel. Energy-efficient cluster-based service discovery for Ubiquitous Computing. *Proceedings of the 11th workshop on ACM SIGOPS European workshop: beyond the PC*, 2004.

[74] T. Sivaharan et al. Cooperating sentient vehicles for next generation automobiles. In *Proc. ACM/Usenix MobiSys Int'l Workshop Applications of Mobile Embedded Systems (WAMES'04)*, 2004.

[75] J. Taheri and A. Zomaya. A Simulated Annealing approach for mobile location management. *Computer Communications*, 30:714–730, 2007.

[76] C. Ververidis and G. Polyzos. Routing Layer Support for Service Discovery in Mobile Ad Hoc Networks. *Proceeding of the Third International Conference on Pervasive Computing and Communications Workshops*, pages 258–262, 2005.

[77] R. Wakikawa, J. Malinen, C. Perkins, A. Nilsson, and A. Tuominen. Global Connectivity for IPv6 Mobile Ad Hoc Networks. *Internet Engineering Task Force, Internet Draft (Work in Progress), July*, 2002.

[78] C. Wewetzer, M. Caliskan, A. Luebke, and M. Mauve. The Feasibility of a Search Engine for Metropolitan Vehicular Ad-Hoc Networks. *IEEE Globecom Workshops*, pages 1–8, 2007.

[79] L. Wischoff, A. Ebner, H. Rohling, M. Lott, and R. Halfmann. SOTIS-a self-organizing traffic information system. In *The 57th IEEE Semiannual Vehicular Technology Conference, VTC*, volume 4, 2003.

[80] O. Wolfson, B. Xu, and H. Yin. Reducing resource discovery time by spatio-temporal information in vehicular ad-hoc networks. In *Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks*, pages 91–92. ACM New York, NY, USA, 2005.

[81] M. Wu, A. Friday, G. Blair, T. Sivaharan, P. Okanda, H. Limon, C. Sørensen, G. Biegel, and R. Meier. Novel component middleware for building dependable sentient computing applications. In *ECOOP04 Workshop on Component-oriented approaches to Context-aware Systems, Oslo, Norway*. Citeseer, 2004.

[82] R. Wunderlich, C. Liu, I. Elhanany, and T. Urbanik. A novel signal-scheduling algorithm with quality-of-service provisioning for an isolated intersection. *IEEE Transactions on Intelligent Transportation Systems*, 9(3):536–547, 2008.

[83] X. Yang, J. Liu, F. Zhao, and N. Vaidya. A Vehicle-to-Vehicle Communication Protocol for Cooperative Collision Warning. *Emergency*, 3(6):114–123, 2004.

[84] P. Zhou, T. Nadeem, P. Kang, C. Borcea, and L. Iftode. EZCab: A Cab Booking Application Using Short-Range Wireless Communication. In *Third IEEE International Conference on Pervasive Computing and Communications, PerCom*, pages 27–38, 2005.

[85] F. Zhu, M. Mutka, and L. Ni. PrudentExposure: a private and user-centric service discovery protocol. *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications, 2004. PerCom 2004.*, pages 329–338, 2004.

[86] A. Y. Zomaya. Hybrid intelligent methods for solving the mobility management problem. In *MobiWac '08: Proceedings of the 6th ACM international symposium on Mobility management and wireless access*, pages 141–141. ACM, 2008.