ENTREGA DE CONTEÚDO EM REDES VEICULARES

FABRÍCIO AGUIAR SILVA

ENTREGA DE CONTEÚDO EM REDES VEICULARES

Tese apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Doutor em Ciência da Computação.

Orientadora: Linnyer Beatriz Ruiz
Coorientador: Antonio Alfredo Ferreira Loureiro

Belo Horizonte
Novembro de 2015

FABRÍCIO AGUIAR SILVA

CONTENT DELIVERY IN VEHICULAR AD HOC NETWORKS

Thesis presented to the Graduate Program in Computer Science of the Federal University of Minas Gerais in partial fulfillment of the requirements for the degree of Doctor in Computer Science.

Advisor: Linnyer Beatriz Ruiz
Co-Advisor: Antonio Alfredo Ferreira Loureiro

Belo Horizonte
November 2015

Aguiar Silva, Fabrício

S586e Content Delivery in Vehicular Ad hoc Networks / Fabrício Aguiar Silva. — Belo Horizonte, 2015
xxvii, 115 f. : il. ; 29cm

Tese (doutorado) — Federal University of Minas Gerais
Orientador: Linnyer Beatriz Ruiz

1. Computação — Teses. 2. Redes de computadores — Teses. I. Orientadora. II. Coorientador. III. Título.

CDU 519.6*22 (043)

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Content delivery in vehicular ad hoc networks

FABRICIO AGUIAR SILVA

Tese defendida e aprovada pela banca examinadora constituída pelos Senhores:

Profa. Linnyer Beatrys Ruiz - Orientadora
Departamento de Informática - UEM

Prof. Antonio Alfredo Ferreira Loureiro - Coorientador
Departamento de Ciência da Computação - UFMG

Eduardo Cerqueira
Prof. Eduardo Coelho Cerqueira
Instituto de Tecnologia - UFPA

Fabrício Benevenuto de Souza
Prof. Fabrício Benevenuto de Souza
Departamento de Ciência da Computação - UFMG

Fátima de LPF Figueiredo
Profa. Fátima de Lima Procópio D. Figueiredo
Departamento de Ciência da Computação - PUC-MG

Jussara Marques de Almeida Gonçalves
Profa. Jussara Marques de Almeida Gonçalves
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 27 de novembro de 2015.

À minha esposa Thais, e às minhas filhas Laura e Clarice.

Agradecimentos

Agradeço a Deus por ter permitido que eu alcançasse mais essa conquista. À minha esposa e às minhas filhas, por estarem sempre presentes e me enchendo de alegria e amor. Sem vocês, nada disso teria sentido. A todos os meus familiares, que me apoiaram em todos os momentos. Um agradecimento especial aos meus pais, que sempre me motivaram a estudar e trabalhar firme para alcançar meus objetivos.

Agradeço à Linnyer, que desde 2001 tem compartilhado comigo os seus conhecimentos e me motivado a buscar um equilíbrio entre trabalho e família. Ao Loureiro, por todo incentivo e contribuições significativas para o desenvolvimento desta tese. A todos os membros da banca, que contribuíram significativamente para a qualidade deste trabalho. Agradeço ao professor Azzedine Boukerche, por ter me recebido em seu laboratório, em Ottawa. À toda equipe do INCT NAMITEC, por todo apoio disponibilizado. Agradeço aos órgãos CAPES e CNPq, que fomentaram esta pesquisa, possibilitando a realização deste trabalho de tese e a interação com outros pesquisadores. Agradeço também a todos os colegas do laboratório Wisemap.

"O período de maior ganho em conhecimento e experiência é o período mais difícil da vida de alguém."

(Dalai Lama)

Resumo

Redes Ad hoc Veiculares (VANETs) estão se tornando uma realidade devido, em parte, ao interesse dos fabricantes de veículos em prover novos tipos de serviços aos seus clientes. Como consequência, as aplicações para esse tipo de rede estão surgindo com o objetivo de tornar o trânsito mais seguro, menos congestionado, mais informativo e prazeroso. Para isso, um dos principais requisitos de tais aplicações é a entrega de conteúdo aos veículos. No entanto, algumas características específicas das VANETs, como topologia extremamente dinâmica, mudanças frequentes na densidade da rede e sua natureza de larga escala, fazem com que a tarefa de entrega de conteúdo seja significativamente complexa.

Normalmente, duas abordagens têm sido usadas para a entrega de conteúdo em redes tradicionais: Redes de Entrega de Conteúdo (*Content Delivery Networks* ou simplesmente CDN) e Redes Par-a-Par (*Peer-to-Peer Networks* ou simplesmente P2P). Porém, as características específicas das redes veiculares sugerem que essas abordagens, como originalmente propostas para a Internet, não são adequadas para esse tipo de rede, e até o momento ainda não está claro como as redes veiculares podem se beneficiar pela utilização dos conceitos de CDN e P2P. Para abordar esse problema, o principal objetivo desta tese é investigar como os conceitos de CDN e P2P podem ser aplicados às VANETs, e propor soluções adequadas a essas redes. Para alcançar esse objetivo, fizemos um estudo detalhado da literatura para entender como os conceitos de CDN e P2P vêm sendo aplicados às VANETs. Em seguida, propusemos um modelo híbrido em que conceitos de CDN e P2P são herdados e adaptados para VANETs. Para auxiliar os projetistas de aplicações, também definimos um arcabouço que engloba os principais componentes a serem implementados para o modelo proposto. Para concluir, com base em um estudo de caracterização de mobilidade realizado, também propusemos e avaliamos duas soluções com diferentes demandas para demonstrar a eficiência do modelo proposto para o desempenho das aplicações.

Palavras-chave: Redes veiculares, entrega de conteúdo, replicação de conteúdo.

Abstract

Vehicular Ad Hoc Networks (VANETs) are migrating from theory to practice mainly due to the great interest of manufacturers to provide new on-road services to their clients. As result, VANET applications are emerging to reality with the objective of making traffic safer, less congested, more informative, and enjoyable. To this end, a fundamental requirement for such applications is the efficient delivery of content. Nevertheless, the particular characteristics of a VANET, such as highly dynamic topology, frequently density variations, and large-scale nature, make the task of delivering content easier said than done.

Usually, two approaches have been used to deliver content in the traditional Internet: Content Delivery Networks (CDN) and Peer-to-Peer (P2P). However, several characteristics of VANETs and their applications suggest that pure CDN and P2P models, as originally conceived for the Internet, are not suitable for them. So far, it is unclear how VANETs can benefit from adopting CDN and P2P concepts.

To address this problem, the main objective of this thesis is to investigate how CDN and P2P concepts can be applied to VANETs and to propose solutions that validate the hypothesis that those concepts are helpful to vehicular applications. To achieve this objective, we first analyze the related studies to find out how the content delivery problem has been tackled in the literature. Based on this study, we define a hybrid model where concepts from both CDN and P2P are inherited and adapted to VANETs. In addition to the model proposal, we also define a framework to guide application designers to develop content delivery applications. Finally, we propose and evaluate two solutions, based on mobility characterization results, to demonstrate that our model is useful for the performance of VANET applications.

Keywords: Vehicular ad hoc networks, content delivery, content replication.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

1.1 Motivation

Vehicular Ad Hoc Networks (VANETs) are migrating from theory to practice mainly due to the great interest of manufacturers to provide new on-road services to their clients [Zeadally et al., 2012]. This kind of network consists of vehicles with on-board wireless communication facilities that are able to establish ad hoc communication with their peers as well as with infrastructure stations [Wang and Li, 2009; Yousefi et al., 2006]. Besides the wireless communication capability, vehicles in a VANET are also equipped with processing, memory, storage, sensors, and visualization units.

VANET applications are emerging to reality with the objective of making traffic safer, less congested, more informative, and enjoyable [Lee et al., 2014; Willke et al., 2009]. With the advance of such applications, comes the need for content delivery solutions, since the efficient delivery of content is a fundamental requirement for most vehicular network applications, such as information, advertisement, and entertainment systems. Information systems deliver to vehicles information such as weather reports, traffic status, parking availability, among others. Advertisements about restaurants, gas stations, and hotels, for example, are delivered to vehicles presenting particular characteristics. Finally, entertainment systems deliver content, such as videos, audios, and images, to be consumed by on-board users.

Nevertheless, the VANET particular characteristics make the task of delivering content easier said than done [Gerla et al., 2014b]. First, vehicles move constantly, causing frequent changes in the network topology. In addition, vehicles may face different density scenarios along their way, such as high density in rush hours or in specific regions, as well as low density in highways or in low traffic hours. Also, vehicular network applications typically operate in large-scale scenarios consisting of many thousands of vehicles. Furthermore, vehicles may face a dispute for bandwidth in order to receive a content within expected quality levels. Finally, infrastructure stations and cellular networks may not be able to cope with all demands. All these issues, together with the fact that users are willing to receive their content quickly, with high quality, and with low cost, bring many challenges to the content delivery problem in such dynamic scenarios. In short, content delivery solutions must be efficient, scalable, and resilient to network topology and density changes.

Usually, two approaches have been used to deliver content in the traditional Internet [Passarella, 2012]: Content Delivery Networks (CDN) and Peer-to-Peer (P2P). CDN solutions rely on the replication of content in the so-called surrogate servers that are strategically placed in the network, and on redirecting a request to the appropriate server. As result, CDN provides high content availability in an infrastructure-based approach. However, it assumes the existence of strategically located stationary servers to replicate content. On the other hand, nodes in P2P solutions cooperate among themselves by offering their resources to their peers, leading to a scalable solution. Usually, nodes can join and leave a P2P network whenever desired, which is a good strategy in terms of scalability, fault-tolerance, and deployment issues. However, content discovery and delivery in P2P networks may take a long time and generate a high network overhead.

Several characteristics of VANETs and their applications suggest that pure CDN and P2P models, as originally conceived for the Internet, are not suitable for them. First, many applications are referred to as push-based, meaning that content should be pushed to the clients even in the absence of a request, like an accident notification, for example. In addition, contacts in VANETs are intermittent, making the establishment and maintenance of end-to-end links very difficult. Furthermore, the deployment of surrogate servers in a large-scale, urban scenario is a costly, time-consuming task. When it comes to content, several entities are potential sources, such as vehicles, Intelligent Transportation System (ITS) solutions, Wireless Sensor Networks (WSNs), mobile users, among others. Moreover, content sources and clients move in considerable speed, causing constant changes in the network topology. Finally, content in many applications is location- and time-dependent, meaning it is valid only inside a region of interest (RoI) during a given period.

1.2 Objective

Given the need for more efficient vehicular content delivery solutions, the hypothesis of this work is defined as: the adoption of a hybrid model that inherits and adapts concepts from both CDN and P2P will improve the performance of VANET applications in terms of content availability, quality, and delivery cost. Therefore, the main objective of this thesis is to investigate how CDN and P2P concepts can be adapted and applied to VANETs, forming a Vehicular Content Delivery Network (VCDN), and propose solutions that validate the aforementioned hypothesis.

The following steps were followed to achieve this objective:

- An in-depth study of the solutions found in the literature has been performed under the content delivery perspective. We have analyzed and organized those studies into a detailed survey of the state-of-the-art in the VANETs content delivery field;
- Based on the survey, we have investigated those studies' benefits and drawbacks, which helped us to map the application requirements. This investigation has led to interesting results such as a classification of VANET applications with content delivery demand, a hybrid VCDN model that inherits and adapts concepts from both CDN and P2P, and a framework to guide designers when developing VANET applications;
- Before implementing solutions to validate our proposed VCDN model, we have first characterized a realistic vehicular mobility trace to demonstrate that some vehicles present special characteristics that may turn them into good replication nodes. The insights from this characterization helped us to design solutions to validate our hypothesis;
- We have proposed and evaluated two solutions with different demands to validate the hypothesis that our hybrid VCDN model, when it follows good practices and insights about mobility models, leads to good performance results, particularly in terms of content availability and delivery cost.

1.3 Contributions

The main contributions of this thesis are summarized as follows:

1. A novel content delivery model that inherits and adapts concepts from both CDN and P2P, as well as a framework to help application designers and developers

on defining the most appropriate architecture and techniques to be adopted by their applications. This model, as well as the framework, will guide designers to develop applications with characteristics such as extensibility, adaptability, and modularity;

2. A survey of VANET content delivery solutions found in the literature organized according to their techniques and architectures. This survey will be of great help to researchers working on this field;

3. Two innovative solutions based on our proposed VCDN model developed to make content available in city-wide and region-wide scenarios. These solutions, in addition to validate our hypothesis, advance the state-of-the-art since they are based on novel methods that consider origin-destination points as input;

4. Characterization results that lead to important insights on how to select appropriate replica vehicles in content delivery applications;

5. A list of potential aspects, and their respective categories, to classify VANET applications under the content-oriented perspective. Based on this classification scheme, which extends the existing ones, application designers will be able to classify their applications prior to deciding which algorithms, architectures, and protocols to adopt, decreasing then the application risk of failure.

1.4 Outline

This document is organized as follows. In Chapter 2, we present a background on important concepts including CDN, P2P and VANET applications, as well as the survey of the state-of-the-art of vehicular content delivery solutions. Next, in Chapter 3 we present a list of aspects to classify vehicular applications under a content-centric perspective, and we propose a model that inherits and adapts concepts from both CDN and P2P and a framework to guide designers on the development of such applications. The characterization of a vehicular mobility trace is presented in Chapter 4. In order to validate our hypothesis, in Chapter 5 we propose and evaluate two solutions by following our proposed model and framework. Finally, in Chapter 6 we present our final remarks and future directions, as well as the list of publications obtained.

Chapter 2

Background

In this chapter we present the fundamental concepts related to the thesis. First, we briefly describe the main characteristics of Content Delivery Networks (CDN) and Peer-to-Peer (P2P). Next, we describe the main categories of VANET applications, presenting examples of how they demand content delivery solutions. Finally, we present an in-depth study of the state-of-the-art in the VANET content delivery area. This survey is the main contribution of this chapter and was very useful to help on directing our research efforts during the thesis development.

2.1 Content Delivery Networks (CDN)

The increase in demand for Internet services, particularly in the context of the Web during the 1990s, has led to network congestion and server overload problems [Hofmann and Beaumont, 2005]. Due to this large demand, content providers started suffering from performance issues and were not able to respond to all requests satisfactorily. To tackle this problem, a new concept called Content Delivery Network (CDN) was proposed in the late 1990s [Peng, 2004; Vakali and Pallis, 2003; Pathan and Buyya, 2007; Pallis and Vakali, 2006]. The basic idea behind CDN is to allocate content replicas in different strategically placed servers, and to redirect a request to the most appropriate server that can respond better to it.

According to the CDN literature [Peng, 2004; Pathan and Buyya, 2007], two major challenges arise when designing a CDN. The first one is the issue of selecting good replica locations and replicating the content, and, thus keeping it up to date. Over this thesis, we refer to this problem as content replication or replica allocation. The second challenge is related to the discovery of the most appropriate replica and the content delivery itself. We refer to it as content discovery and delivery along the text.

These issues are the two fundamental building blocks that compose a CDN system, and are even more challenging in such dynamic environments as VANETs.

Usually, Internet content replication is done based on geographic locations. It is expected that users in Europe would have a better performance when downloading content from servers located in that continent than in America, for example. Therefore, it is reasonable to place replicas in servers located all over the world according to the expected demand. Since surrogate servers are fixed, as well as most part of the Internet infrastructure, there is peak on investment to select the most appropriate replica places. After that, there is no need to change places constantly.

Because content replica locations are known, the content discovery and delivery tasks are straightforward. Basically, a client sends a request message to the original content provider that knows where all of its replicas are located. The server then responds to the client with the address of the most appropriate replica to attend its request. The selection of the replica is a difficult task, since not only the physical distance, but also the network distance and the server current load, must be taken into account. The client then exchanges messages with the surrogate server. The most adopted technique to content discovery and delivery is the one of Dynamic Name System (DNS) redirection. In this technique, the DNS server is responsible of responding to the client the IP address of the surrogate server that will attend it.

The benefits of CDN to the Internet applications performance are clear and well-known. However, the same can not be stated when it comes to VANET applications, since this kind of network presents particular characteristics that differ it from the Internet. Thus, further investigation must be performed in order to identify how CDN concepts could be beneficial to VANET applications.

2.2 Peer-to-Peer Networks (P2P)

A P2P network [Blair et al., 2012; Lua et al., 2005; Androutsellis-Theotokis and Spinellis, 2004; Parameswaran et al., 2001] is a scalable self-organizing system in which nodes cooperate to form a content delivery network. Nodes in a P2P system are referred to as peers since they may act as clients, when downloading content from others, and servers as well, when providing content to other peers. Usually, the peers are personal workstations of ordinary users (i.e., personal computers) connected through a network, instead of powerful servers located in data centers. All peers contribute parts of their resources (network bandwidth, storage, etc.) and usually have the same capacity and responsibility.

P2P networks came into reality back in 1999 in the U.S., when a significant number of ordinary users started using the Internet. The number of P2P systems has increased as more users, all over the world, also were attracted to the Internet. The first success application was Napster [Carlsson and Gustavsson, 2001], which was conceived for music sharing. Due to copyright problems, Napster was considered illegal and closed by U.S. authorities. However, the Napster case was very important to bring large-scale P2P networks into reality.

Differently from CDN, an interesting fact about P2P networks is the needless of a dedicated infrastructure. Any peer can participate of the content delivery task, and usually no central controller is required. The only technical requirement is that all peers should run a specific program to be prepared to join a P2P system. P2P networks are scalable by nature, since their upload capacity increases with the download requirements, since all nodes may participate and help on the delivery.

Given that personal workstations are not expected to be online 100% of the time, since their owners may turn them down and the machines may failure as well, there is no guarantee of individual resources availability. Thus, P2P systems are prepared to increase the probability of content availability by keeping copies of it on different peers that requested it recently. Therefore, P2P networks are fault-tolerant because failures in a peer is suppressed by others.

Basically, there are two approaches to implement P2P networks: structured and unstructured. The structured strategy relies on distributed data structures to control the network topology and the content placements. These structures are also used when a client is searching for a content. This leads to efficient algorithms for searching for content, with the high cost imposed to build and maintain those distributed structures. On the other hand, in the unstructured strategy the peers join and leave the network by following simple rules in an ad hoc manner. This self-organizing network is, therefore, resilient to failures in the peers. However, searching for a content may generate a significant overhead in terms of exchanged messages and a long end-to-end delay, since there is no controller entity to help on this task.

The unstructured approach is more appropriate to VANETs, given their highly dynamic topology and, then, the high cost involved to build and maintain the distributed structures. However, further investigation should be conducted to propose efficient solutions for content discovery and delivery with the objective of providing a high content delivery service to their users.

2.3 Information-Centric Networking (ICN)

ICN is a novel communication paradigm in which content is requested by clients from the network instead of from particular hosts [Carofiglio et al., 2013]. The most promising characteristic of ICNs, when compared with the traditional content delivery solutions such as CDN and P2P, is that a request is propagated into the network containing the name of the content a user is interested in, instead of the address of the content provider. In this approach, referred to as name-based content delivery, the requests are then routed towards the hosts that contain such content according to its name.

In addition to the adoption of the name-based approach, ICNs also take advantage of in-network caching. This functionality assumes that routers are able to store in their cache, for a period of time, content they relay. The objective is to reduce the delay of finding a content by keeping copies of it in different routers.

In the recent years, researchers started paying attention to the ICN field, including its application to VANETs [Bruno et al., 2015; Grassi et al., 2014; Amadeo et al., 2013]. However, in contrast with CDN and P2P, there is no consensus among researchers on the detailed architecture, protocols and services of an ICN. This debate is even less advanced when it comes to VANETs. It is expected that ICN will play an important role on content delivery networks for VANET applications, given the dynamic characteristic of this type of network. Therefore, in this thesis we pay attention to this observation by making our proposals compliant with the ICN paradigm.

2.4 VANET Applications

With the objective of identifying how CDN and P2P concepts can be adapted and applied to VANETs, we need first to study the applications' demands. Basically, VANET applications are organized into four groups: Safety, Information Systems, Advertisements, and Entertainment. Next, we present examples of such applications by describing their demand for content delivery solutions.

Safety Safety applications aim at making traffic safer and helping users to avoid accidents by alerting them about imminent dangerous situations. Usually, content required in these applications is small, localized, and delay-sensitive since accidents may happen in the order of microseconds.

For example, when a vehicle is approaching rapidly or intends to change lane suddenly, alert messages should be delivered to inform other vehicles that can make decisions to avoid accidents. Another example relates to intersection traffic, which is a common area of accidents. Thus, vehicles approaching an intersection must be notified about traffic conditions and vehicles expected to cross the intersection simultaneously. This helps users to be aware of miss-judgement about traffic situations.

Information Systems These applications aim at informing users about situations that are of interest to them. Many different information systems are useful for users, as exemplified as follows.

Based on historical traffic information, vehicles may identify when traffic is not flowing as expected for the current road and time, inferring then a traffic jam situation. Thus, this situation may be informed to other vehicles moving in that direction so they can, for example, take an alternative route whenever possible. Usually, content for such scenarios is small and must be delivered to users moving towards the traffic jam area within a restricted delay in the order of a few seconds.

In the occasion of an accident, usually the rescue team, including fire fighters and paramedics, moves to the accident area with little or none information about the accident. For example, information about the number of victims and their health status and whether there is a fire risk or not, may not be available, making difficult to plan the rescue prior to arriving at the accident area. Thus, vehicles equipped with cameras may transmit a real-time video stream showing the accident area to the rescue team. Content in such case is large and must be requested by the rescue team or other interested parts. The video should be delivered with a minimum accepted delay to be reproduced correctly.

Usually, finding a parking spot takes a significant amount of time. Hence, users should be informed about parking availability on their final destination. To help on this task, vehicles may be able to monitor and identify free parking spots without external help. Usually, parking availability content is small and should be delivered to users that request it or not, depending on the application requirements. Also, such content must be delivered within a deadline that will allow the vehicles searching for a spot to park at the informed place.

Other information systems are also of interest to VANETs. One example is the delivery of weather reports referred to the users' route or their final destination. This way, users will be aware of rainy or snowy conditions, which require driving attention from them. Another example is the delivery of information about gas station locations on his route when fuel level is below a threshold. The same should be done to inform users about mechanical problems, indicating nearby places for maintenance. Finally, to help tourists, leisure information about the arriving city may be delivered according the the users' personal interests. Information about tourist sites, restaurants, and hotels, are among the examples of such content. Content for such applications may differ in its characteristics depending on the application requirements. For example, it should be small (e.g., only textual information) or large (e.g., multimedia content showing a tourist place or the weather situation). Also, such content may be requested by users or delivered to them when it is expected to be useful. Moreover, they may present different delay constraints.

Advertisements Users at specific locations may receive advertisements about restaurants, hotels, gas stations, and general shopping places that are of interest for them. Usually, advertisements may be sent to users based on their contextual and personal interests. Users may require specific advertisements about something, or may receive them without requesting when the application senses that it should be of interest to the users. Advertisements may be simple text messages, or elaborated content such as images and videos. Also, advertisements should be delivered in the most adequate period (i.e., when users need them).

Entertainment Another promising application for VANETs is for users entertainment. Entertainment content is usually available from Internet servers to be downloaded and consumed by on-board users. Examples of such a content include videos from YouTube, news web pages, blog pages, video-on-demand content from Netflix, or any general multimedia content of interest to the users. Entertainment content is very particular to each user interests and profile. So, usually they are pulled from users from the providers whenever required. Also, depending on the user interest, content should be delivered to be consumed immediately, or to be consumed after arriving at the final destination. Then, delay constraints must be defined accordingly.

All the examples mentioned require different types of content to be delivered to vehicles. In the next section, we present how studies from the literature have been addressing the content delivery issue for VANET applications.

2.5 Vehicular Content Delivery Solutions

Basically, there are two major challenges when designing content delivery solutions. The first one is the issue of selecting good places to replicate content. The second challenge is related to the discovery of the most appropriate server, either origin or surrogate, and the content delivery itself. These issues are the two fundamental building blocks that compose a content delivery system, and are even more challenging in such dynamic environments as VANETs.

The studies found in the literature adopt different terms to refer to the same concepts. Thus, we organize the main concepts used and their respective terms in Table 2.1. This taxonomy table helps on the reading of this thesis, as well as those of other related work. When describing and discussing specific studies, we use the same terms as their authors have done to facilitate the comprehension of further references to the original study.

Concept Description	**Related Terms**
Selection of specific nodes to act as temporary content providers	Replica allocation; Content Replication; Replica selection; Replica placement;
Nodes selected to act as temporary content providers	Replica keepers; Carriers; Surrogate servers; Mobile storage; Replica nodes; Bearers;
Fixed infrastructure station placed on the roads	Road-side unit (RSU); Base Station (BS); Access Point (AP); Infostation;
Node interested in some content	Client; User; Requester; Consumer; Subscriber; Downloader;
Node providing content	Server; Provider; Publisher;

Table 2.1. Taxanomy for Content Delivery in VANETs.

2.5.1 Challenges

In the past few years, the content delivery problem started being explored by the research community in the VANET field. This was due to the rapid increase in VANET development, also leveraged by vehicle manufacturers. However, it turns out that traditional solutions, as originally conceived for the Internet, may not be applied to VANETs due to their differences when compared to traditional networks. As discussed in the following, the specific characteristics of VANETs make the development of content delivery solutions even more challenging, but it is still an open issue.

First, VANETs present a highly dynamic topology, posing increasing difficulties in selecting and maintaining the replicas. Vehicles are in constant movement at different speeds and acceleration. This leads to constant changes in the network topology, because contacts among vehicles are continually established and terminated. Thus, the most appropriate vehicles to act as replicas may also change over time. In contrast, surrogate servers in the Internet are stationary and strategically placed where they are expected to be useful, based on content demand, historical facts, and expectation. Therefore, there is no need to constantly change the replica places.

In addition, high vehicle mobility suggests that several different server replicas will be required to complete a delivery. Contact between vehicles may not last enough to deliver full content. In [Uppoor and Fiore, 2012], the authors, during a large-scale mobility trace from the city of Cologne, in Germany, showed that most contacts between vehicles lasts no longer than 15 seconds. This amount of time may not be sufficient for delivery, in which case a vehicle will require many providers to receive content. In fact, this makes the content delivery task even more difficult in such dynamic scenarios.

For some applications, content refers to specific locations, and must be delivered only to those vehicles that are passing by or travelling in the direction of those respective locations. Some content, such as video of an ongoing traffic jam, may only interest vehicles in specific regions. Thus, the delivery process may be aware of this situation and make decisions based on that. In addition, content may only be valid for a period of time (e.g., during a traffic jam). Outside this validity time frame, it has no utility at all. Two extra variables must then be added to the content delivery system: space and time. This poses even more challenges for content replication and delivery.

Another issue is the difference in network density over the duration of time and space, which increases the cost of selecting the replicas and delivering content to users. Network density may differ significantly, depending on the time of day (e.g., peak hours or late at night) and the region (e.g., downtown or rural regions). This requires content delivery to be aware of and adaptive to different network density scenarios. Some authors have been working on solutions to measure and predict the traffic intensity in the roads, and consequently the network density, such as [Younes and Boukerche, 2015]. However, a lot of work should be done on this field. Therefore, the density variability issue increases the complexity of a content delivery system, particularly in large-scale environments.

Finally, VANET solutions are intended for deployment in large cities that have hundreds of thousands of vehicles on the roads. Furthermore, connected vehicles are expected to be integrated to the Internet of Things (IoT) [Piro et al., 2014; Borgia, 2014; Gerla et al., 2014a]. In such complex scenarios, content may be provided by and to a number of entities other than vehicles (e.g., intelligent semaphores, smart cameras, mobile devices, drones) in a large-scale, heterogeneous architecture. Thus, vehicular content delivery systems must also be efficient, resilient, and scalable.

All of these issues imply that many challenges exist in the vehicular content delivery network field. To help address these challenges, it is important to have a background in the existing solutions and their characteristics. In this chapter, we survey the studies found in the literature that propose replica allocation (Section 2.5.2) and content delivery (Section 2.5.3) solutions applied to VANETs.

2.5.2 Replica Allocation Solutions

The idea behind a content delivery solution is to keep content replicas in nodes close to the clients, and to then instruct them to use the replicas instead of the original server itself. One of the main challenging issues, when it comes to VANETs, is the selection of appropriate vehicles or RSUs to act as replica nodes. In this section, the studies that propose solutions to the replica allocation problem applied specifically to VANETs are described and analyzed. Some studies applied to MANETs or cellular networks considered relevant to VANETs scope are also presented. The solutions are analyzed from the architecture perspective that defines how their entities are organized and how they communicate among themselves. The decision of which architecture to use plays an important role on the performance of replica allocation.

From the VANET architecture perspective, there are basically three approaches adopted to the replica allocation process: centralized, distributed (infrastructure-based or infrastructure-less), and hierarchical. The studies found in literature for each of these categories are analyzed in the following.

2.5.2.1 Centralized Approach

In the centralized approach, the decision regarding the selection of vehicles as replicas is made by a centralized entity (e.g., RSU, AP, and Internet server). The centralized entity is expected to have a high computational capacity in terms of memory and processing, a constant energy source, and a wide bandwidth. In addition, the centralized server may take advantage of a broader view of the network. However, care should be taken when allocating replicas in a large-scale scenario.

The centralized solutions usually require a significant amount of knowledge of network topology and status, as the studies described. In MobTorrent [Chen and Chan, 2009], the replica nodes are selected for each content request. Based on the expected contact graph, the provider AP replicates chunks of the content to other APs, as well as to other vehicles, with the objective of maximizing the amount of data transferred to the requesting vehicle. The selection of carrier vehicles depends on the vehicles' movement direction and expected encounter time with the requesting vehicle and the APs. This solution requires a precise vehicle encounter prediction and APs position, which may not be accurately available in large-scale dynamic networks. Another study proposes Push-and-track [Whitbeck et al., 2012], which keeps track of the nodes that have already received content, and decides whether to re-inject new replicas into the network. Through exhaustive simulation in a realistic mobility scenario, the authors showed that random selection of replica nodes outperformed other strategies like entry time, position, and connectivity-based approaches. The major drawback of both solutions is the requirement for vehicle mobility behavior or for the network connectivity graph, which may be costly to obtain with acceptable accuracy. Finally, an optimization solution is described in [Bruno et al., 2014], where content is placed into RSUs based on its demand and popularity, with the objective of maximizing the content availability. In contrast to other existing solutions, this proposal assumes the adoption of the Information-Centric Network (ICN) model [Liu et al., 2014; Grassi et al., 2014; Amadeo et al., 2013; Bai and Krishnamachari, 2010], in which the content search and delivery consider the content's name instead of its physical location.

Other studies propose solutions to deliver content from one fixed source station to a destination station. Thus, a vehicle is selected to be the content carrier from the source to the destination. On-Time [Acer et al., 2011] is a routing protocol for bus transportation systems. The objective is to deliver content from one point to another, using buses as carriers. Based on the scheduled stops of each bus, an algorithm that tries to maximize the delivery probability within a given period is used to select the best carrier bus. In a similar manner, but one that considers all vehicles in a highway scenario, OVS-OBRM [Khabbaz et al., 2012] selects a vehicle to be the carrier of content that must be delivered from one RSU to another. Thus, the authors propose the computation of the residual travel time (the time taken for each vehicle to reach the target RSU). The vehicle in the vicinity of the source with the smallest residual travel time is then selected as the carrier. The main drawback of these solutions is the assumption that both source and destination are fixed entities.

Other studies evaluate centralized approaches for selecting target nodes to adopt opportunistic communication in order to offload the cellular network. Although not directly applied to VANETs, they propose innovative and interesting solutions and are described in this thesis, since they could be adapted to VANET scenarios. Opp-Off [Han et al., 2010, 2012] selects an initial set of target users to exploit opportunistic communication for dissemination in cellular networks. The idea is to maximize the number of users reached and to minimize the amount of cellular communication. However, since this is a NP-hard problem, the authors propose and evaluate three algorithms: random, greedy, and heuristic. The heuristic takes into account the expected mobility pattern. The greedy approach achieves the best results, as it approximates the best case of target selection. However, it requires the user's mobility behavior, which may not be easy to obtain. Similarly, TOMP [Baier et al., 2012] is a cellular opportunistic offloading strategy that selects some mobile devices as the initial target set. This set of mobile devices is then responsible for opportunistically disseminating the information to its peers. The target set selection in TOMP takes into account the position and speed of each mobile device. The mobile devices that are expected to encounter a higher number of peers are selected as part of the target set. Other existing solutions [Thilakarathna et al., 2013, 2014; Barbera et al., 2014] rely on social network metrics to select appropriate users of smartphone devices to replicate content in order to offload the core cellular network. The overall idea is to exploit the knowledge of users interaction to find the most appropriate users that would lead to good offload results. All those solutions are lacking in the scalability area since they require a large amount of information and complex algorithms to run.

In general, the centralized solutions achieve good coverage results with respect to their purposes. However, they do not scale well due to their computational complexity. Additionally, they require a significant amount of up-to-date information to operate properly. In fact, most proposals for replica allocation applied to VANETs consider distributed algorithms that may or may not take advantage of infrastructure stations, as described in the following.

2.5.2.2 Distributed Approach

In the distributed approach, the replica selection uses distributed algorithms and protocols that only consider localized information. Some distributed solutions are infrastructure-less, with the decisions made only by the vehicles themselves. Others, on the other hand, are infrastructure-based, since they take advantage of infrastructure stations. We describe the infrastructure-based solutions next, followed by the infrastructure-less.

Infrastructure-based distributed solutions In the infrastructure-based distributed approach, decisions are made in a distributed fashion with help from infrastructure stations. This can be considered a reasonable assumption. It is expected that VANET scenarios will be covered by infrastructure communication capabilities, for example, traditional cellular networks or even V2I dedicated short-range communication using recent standards like the IEEE 802.11p [Department, 2010]. The main advantage of the infrastructure-based distributed approach is the capability of using a computational system with a broader view of the network without having to organize a hierarchy among vehicles. Some studies in the literature have already exploited this advantage in their proposals on VANET replica allocation procedures, as described in the following.

In Figaro [Malandrino et al., 2012], content management is performed by brokers, which are entities running on infrastructure computational systems. Each broker is responsible for a set of users, and receives advertising of their local content from them. By having a complete view of the content availability and requests in its region, a broker is able to decide which content must be kept as a replica by the user that has received it. A broker also decides which content must be replicated and where, based on its popularity. The disadvantage of this solution is the overhead caused by content report messages, since vehicles move constantly and rapidly from one broker coverage to another. Similarly, VTube [Luan et al., 2014, 2011] explores the RSUs, described as Road-side Buffers by its authors, to replicate content where it is more likely to be requested. To this end, VTube takes advantage of the content popularity to draw a distributed solution for increasing the expected download rate.

Some proposals take advantage of the expected connectivity graph built by infrastructure stations to decide how to select carriers and schedule a content download. In TEG-PW [Malandrino et al., 2014], RSUs keep track of content availability. Based on mobility predictions, an RSU formulates a linear programming optimization problem to schedule content delivery by selecting the carriers to pre-fetch content chunks. These carriers are selected based on the encounter time-expanded graph prediction, which is built by a traffic manager system using mobility information messages from vehicles. Each RSU updates its contact prediction map based on the local contacts it observes. Similarly, in [Trullols-Cruces et al., 2012], the APs maintain contact maps based on overhearing the messages exchanged among vehicles. The contact of two vehicles is predicted based on historical moments of contact that occurred between two vehicles that have moved in a similar pattern to the current ones. The contact map is exploited by an AP to select the vehicles required to carry content that should be downloaded by other vehicles, using an estimated encounter prediction. Several algorithms based on contact probability are evaluated through simulations. The main drawback of the described solutions is their requirement for the connectivity graph and the future contact prediction, which are costly to maintain in dynamic networks such as VANETs.

A hybrid approach is also explored in two studies [Leontiadis, 2007] and [Leontiadis et al., 2009a], that focused on keeping replica content in the vicinity of a region of interest. The content is transmitted to a region by *Infostations* located in that region, as well as by vehicles passing by or expecting to pass by. Vehicles are selected as replicas according to their movement characteristics, which are used as input for utility function. This is done by a current content carrier that checks whether one of its neighbors may be a better carrier, and, in a positive case, transfers the content to its neighbor. One drawback of this solution is the overhead caused by periodic messages sent by vehicles containing their mobility status to be used as input for the utility function.

In general, infrastructure-based distributed approaches are scalable, since they take advantage of infrastructure stations as well as distributed algorithms. However, they require more complex solutions because of their distributed fashion. Furthermore, the infrastructure stations must be strategically placed to cover the application scenario.

Infrastructure-less distributed solutions In this approach, the decision pertaining to which vehicle should keep a replica of content is made without the help of any infrastructure station, and is based only on local information. On one hand, solutions in this group tend to scale well to larger scenarios. On the other hand, the limited information used as input for the algorithms may lead to a poor replica allocation. Some solutions, as described in the following, follow the infrastructure-less distributed approach.

Some studies propose schemes similar to caches, in which content is stored for a period of time only in vehicles that have received it. Although caching schemes differ from replica allocation [Padmanabhan et al., 2008], some of them are worth mentioning in this thesis since they may be used in conjunction with allocation schemes to improve content availability in vehicular content delivery networks. Furthermore, we can think of a cache as a replica self-allocation process, in which the vehicles themselves are able to apply a strategy to decide whether or not to act as a replica. Other caching schemes are omitted here and can be found in specific surveys, such as [Padmanabhan et al., 2008].

In InfoShare [Fiore et al., 2005], replicas are kept by vehicles for a period of time after they receive requested content. All content received is cached, and no further information is used to help with the decision of caching. Similarly, InfoCast [Sardari et al., 2009] proposes that vehicles are selected as carriers when they successfully receive content. After that, the vehicle periodically broadcasts the content to its neighbors. On the other hand, Hamlet [Fiore et al., 2011, 2009] is a cooperative caching scheme in which nodes estimate their neighbors' caching to decide which content to keep in their own cache, and for how long. This is done through query and response message overhearing, which allows nodes to be aware of content belonging to their neighbors. In this way, a diversity of content is expected in the vicinity, increasing data availability. In general, the disadvantage of the caching schemes is that a large number of unnecessary replicas may be created. This makes schemes difficult to maintain, and may cause overhead and congestion on the network. However, together with replica allocation schemes, caches may help increase content availability.

Another strategy for VANETs is the adoption of peer-to-peer (P2P) swarming protocols such as BitTorrent [Cohen, 2003]. In SPAWN [Nandan et al., 2005], the vehicles retain their downloaded content, and cooperate among themselves to improve data availability. Unlike the traditional centralized approach, peer discovery in SPAWN is done in a distributed manner by gossip messages exchanged among neighbors. The disadvantage of this solution is that it may not be possible to find a content replica close to the requester; also, depending on the delay in time it takes for a request to reach a provider, the network topology may have changed significantly, which increases delivery cost.

In the approach presented in [La et al., 2012], each node retaining content decides, based on its capacity and workload, whether the content should be dropped or replicated to other nodes. The replication is performed when the node decides it cannot sufficiently attend to all demands in its neighborhood. The placement of replicas is based on the random walk diffusion mechanism, in which content moves randomly from one replicated node to another, considering only 1-hop communication. The replica content movement occurs after a period of storage time. A similar strategy is used in [Khaitiyakun and Sanguankotchakorn, 2014], in which vehicles are selected to keep content replicas based on lower-layer information. More specifically, a subset with higher coverage areas of the Multi-Point Relay (MPR) nodes, identified by the routing protocol, are selected as replica vehicles.

An interesting research focus is related to *geocast* delivery. In *geocast* applications, each content is specific to certain regions of interest, and only vehicles occupying or moving into that region must receive it. The pure distributed approach for replica allocation fits well with the *geocast* demand because the decision may be based only on information local to the region of interest. When it comes to replica allocation for such *geocast* demands, the idea is to select vehicles to which content can be transmitted, in the region of interest; thus, the majority of vehicles passing by will receive the content. Basically, what distinguishes the studies in this field is the practice of selecting vehicles in the best and least costly way possible. In the following, the main *geocast* solutions are presented.

In [Maihöfer et al., 2005], a *geocast* dissemination strategy elects a node to keep replicas of data inside a region of interest. The election process is based on the length of time each vehicle will stay inside the region. To make the election process less costly, more than one vehicle may be elected as replica keeper. A new election process begins when the previously selected node is no longer appropriate because it has left the region. In RADD [Kumar et al., 2015], the vehicles in a region decide on the best replica, based on their number of connections, velocity, communication range, and the number of replicas in the vicinity. They exchange messages containing their parameters, and the one with the highest index is selected as the replica.

Another *geocast* solution is ARM [Borsetti et al., 2011], a framework that also elects good content carriers in a distributed way. This process of selection is based on the following information: carrier distance from the target central point, the angle between vehicle direction and the target central point, vehicle speed, and the target area size. Only one node is elected for each instance of content, and a new election round is begun when the node is no longer deemed appropriate to act as a carrier. The main drawback of this solution is the overhead caused by messages exchanged in the selection process, as well as by monitoring when a new election must take place. Finally, Linger [Fiore et al., 2013] is a protocol used to transmit information in a geographic region of interest. To this end, the authors propose an index that is computed locally by vehicles and that takes into account the distance to the center point of the region of interest, the angle of vehicles (relative to the center point), and speed. This solution, in contrast to the others, does not require the knowledge of vehicle trajectory. However, an overhead is created by the message exchanges required to compute the index values and to select the carriers.

The pure distributed (i.e., infrastructure-less) solutions for replica allocation use local information to decide which vehicles are more appropriate to act as content carriers. Some of the studies follow a cache-based approach, while others consider vehicle trajectory knowledge or current mobility patterns. Another approach is to let vehicles themselves compute the local index and decide which are more appropriate to act as carriers. In general, these indices are computed based on vehicles' information, such as position, speed, and direction. The distributed infrastructure-less solutions are scalable, as they require only local information. However, they are more complex and require more message exchanges among neighbors, which may cause network overhead. This kind of solution also tends to be fault tolerant, as failures may be locally identified, and can then be solved as soon as possible.

2.5.2.3 Hierarchical Approach

Most studies proposing cluster-based replica allocation algorithms have their focus on MANETs [Sharma et al., 2010]. Despite being proposed specifically for MANETs, some of these studies are discussed in this thesis because they describe innovative solutions.

When it comes to VANET scenarios, hierarchical architectures are not well explored due to their highly dynamic vehicular topology; this may lead to a high cost for the maintenance of a hierarchical structure. However, it is important to describe how MANET solutions work, which will give insight into their applicability for VANETs.

In [Huang et al., 2003], the authors explore the use of mobility behavior to organize groups. Based on the motion behavior of nodes, they propose DRAM, a decentralized algorithm used to organize clusters in which nodes have similar motion behavior. Replicas of all content are then allocated based on their access frequencies and the derived allocation units. To avoid the overhead of flooding messages, the proposed solution does not require the knowledge of global network connectivity.

Distributed Hash Tables (DHTs) is a well-known data structure used to create indices for content search. In [Martin and Hassanein, 2005], a Distributed Hash Table Replication (DHTR) system is proposed. In DHTR, the cluster heads keep information regarding the cluster node replica content in a local replica cache. Furthermore, a global replica cache keeps information regarding which content is maintained by each cluster member. The members of the cluster monitor their cluster head status and start a re-election process when the cluster head is no longer available.

When a group of mobile users intends to download the same content, they may cooperate to reduce bandwidth consumption and improve data availability. This problem is tackled in [Stiemerling and Kiesel, 2009] and [Stiemerling and Kiesel, 2010], in which mobile nodes in the proximity elect a node to be the central controller. The controller node is responsible for coordinating which chunks of data each mobile node should download from the Internet; this decision is based on local demands and throughput measurements. The idea is to increase the probability of fetching required content within the deadline.

FCD [Stanica et al., 2013] is one of the few content delivery solutions that focuses on content flowing from vehicles to infrastructure servers. The authors propose a scheme to select a small number of vehicles to receive the collected data from other vehicles in a region; this scheme then proposes to use cellular communication to deliver all data to the infrastructure. Topology metrics (i.e., node degree and assortative organization) computed locally by vehicles over time are used to decide which vehicle will be responsible for each region.

Slinky [Kawadia et al., 2011] is a content networking protocol that organizes the network into communities and keeps content replicas in each community. Slinky also defines a scheme to replicate content across the communities. Community formation is achieved by adopting a distributed version of the greedy approach for the minimum domination set solution; it requires only local knowledge (a small number of hops) of the network topology.

In VANETs, it is expected that the movement of some vehicles will follow the same behavior due to speed constraints, road capacity, and daily activity cycles in urban scenarios. For example, in [Uppoor and Fiore, 2012], the authors showed, based on a realistic large-scale mobility scenario, that there are some patterns in mobility flows that operate according to different periods of the day. This group mobility behavior can be exploited through the proposal of hierarchical replica allocation schemes intended for VANETs.

When it comes to vehicular networks, hierarchical replica allocation solutions are not yet well explored. The main drawback of hierarchical solutions is high maintenance cost, which is not suitable for the large scale and highly dynamic topology of VANETs. However, based on the fact that vehicles may present a group mobility behavior, it is a good idea to exploit this issue when proposing hierarchical solutions. If the cost of organizing and maintaining clusters could be reduced, this approach may be very useful to VANETs.

2.5.2.4 Remarks

Based on the studies described above, we argue that each architectural approach has its advantages and disadvantages, as summarized in Table 2.2. In general, it is possible to make some important observations that concern the architecture adopted thus far in VANET replica allocation solutions. The centralized approach takes advantage of a global view of the network, which enables the adoption of good allocation graph algorithms. However, it is lacking in terms of scalability, since it requires a large amount of up-to-date and accurate data to operate properly. In contrast, the pure distributed approach scales well. However, this approach is complex, since it requires a significant overhead on the network, and cannot take advantage of a broader view of the network. The distributed approach that requires help from infrastructure stations can balance those drawbacks by reducing the overhead, increasing the scalability, and taking advantage of a broader view of the network. However, this approach requires infrastructure stations placed in well-planned areas, which increases the deployment cost. The main drawback of the hierarchical approach is the cost needed to organize and maintain clusters; this is even more significant in highly dynamic networks, including VANETs. By the time the clusters are formed, the cluster heads can use a broader view of the network to operate.

In addition to the architecture adopted, each solution can also be categorized according to its input data, which is the data it requires to operate and allocate replicas.

Approach	**Advantages**	**Disadvantages**
Centralized	Global vision of the network Takes advantage of topology-aware algorithms Does not require complex distributed algorithms and protocols	Single point of failure Does not scale well Topology and other information may be out-of-date
Distributed	Does not require a global processing unit Easy access to up-to-date local information	Complexity of distributed algorithms and protocols Only partial vision of the network
Hierarchical	Can adopt topology-aware algorithms for each cluster There is a cluster head to coordinate the activities	Cost to organize and maintain the clusters

Table 2.2. Replica Allocation Architecture Approaches

The solutions' input data can be classified into four categories: *Network Topology*, *Expected Network Topology*, *Vehicle Information*, and *Content Demand*. The definitions of the first two are straightforward. *Network Topology* refers to the current graph representing the vehicles, the RSUs, and their contacts. On the other hand, the *Expected Network Topology* refers to the graph representing the network topology in the future; in other words, it represents the predicted topology graph. The *Vehicle Information* input data may refer to different aspects, depending on the solution. In general, it refers to the vehicle speed, position, and direction. Finally, *Content Demand* refers to the popularity of content, which may indicate the probability of a content to be requested.

Each solution can also be categorized according to its solution basis. The main solutions presented in this thesis can be classified into three different solution basis: *Graph-based*, *Index-based*, and *Self-allocation-based*. Solutions in the *Graph-based* class adopt graph algorithms (e.g. maximum network flow, minimum domination set, among others) to select replica vehicles that are expected to achieve high coverage and high delivery rates. In general, the *Index-based* solutions use their input data to compute a comparable value that is used to select the most appropriate replica vehicles. Finally, *Self-allocation-based* refers to solutions in which a vehicle itself is responsible for deciding whether or not to keep a local replica. The most relevant solutions found in the literature and discussed previously are summarized in Table 2.3. For each solution, this summary presents its input data required for operation, its solution basis, and some comments.

Solution	**Input**	**Solution Basis**	**Comments**
Centralized			
MobTorrent [Chen and Chan, 2009]	Expected Network Topology	Graph-based	Depends on prediction accuracy
Push-and-track [Whitbeck et al., 2012]	Network Topology	Graph-based	Cost to keep track of covered vehicles
On-Time [Acer et al., 2011]	Expected Network Topology	Graph-based	Assumes fixed source and target
OVS-OBRM [Khabbaz et al., 2012]	Vehicle Information	Index-based	Assumes fixed source and target
Distributed Infrastructure-based			
Figaro [Malandrino et al., 2012]	Network Topology, Content Demand	Graph-based	Overhead caused by content advertisement messages
VTube [Luan et al., 2014]	Content Demand	Graph-based	Complexity
TEG-PW [Malandrino et al., 2014]	Expected Network Topology	Graph-based	Depends on prediction accuracy
Cooperative [Trullols-Cruces et al., 2012]	Expected Network Topology	Graph-based	Depends on prediction accuracy
Hybrid P/S [Leontiadis et al., 2009a]	Vehicle Information	Index-based	Overhead caused by report messages
Distributed Infrastructure-less			
InfoShare [Fiore et al., 2005]	Vehicle Information	Self-allocation based	Overhead caused by queries broadcast
Hamlet [Fiore et al., 2011]	Content Demand	Self-allocation based	Overhead caused by queries broadcast
InfoCast [Sardari et al., 2009]	Vehicle Information	Self-allocation based	Overhead caused by content broadcast
SPAWN [Nandan et al., 2005]	Vehicle Information	Self-allocation based	Overhead of replica discovery
Abiding Geocast [Maihöfer et al., 2005]	Vehicle Information	Index-based	Overhead caused by the selection process
RADD [Kumar et al., 2015]	Vehicle Information	Index-based	Overhead caused by the index calculation process
ARM [Borsetti et al., 2011]	Vehicle Information	Index-based	Overhead caused by the index calculation process
Linger [Fiore et al., 2013]	Vehicle Information	Index-based	Overhead caused by the index calculation process
Hiearchical			
DRAM [Huang et al., 2003]	Content Demand	Index-based	Overhead due to cluster management
DHTR [Martin and Hassanein, 2005]	Vehicle Information	Index-based	Overhead due to cluster management
FCD [Stanica et al., 2013]	Network Topology	Graph-based algorithms	Assumes fixed target
Slinky [Kawadia et al., 2011]	Network Topology	Graph-based algorithms	Scalability issues

Table 2.3. Replica Allocation Solutions Summary

We also outline below some important characteristics regarding the solutions' foundational aspects. First, current and expected network topologies provide very useful insights; they enable the scheduling of delivery so that clients receive different parts of content from different replica sources, depending on their trajectory and expected encounters. However, they require a significant amount of up-to-date and predicted information regarding vehicles, traffic conditions, traffic light schedules, and so on. The acquisition of this information is a difficult and costly task, especially on highly dynamic networks such as VANETs. On the other hand, index-based solutions, in which vehicles compute indices based on local information, require less computational effort. Furthermore, if well-defined, the computed index may lead to good replica selection. However, these solutions require complex distributed protocols to control the replica allocation process; they also lack a broader view of the network. In general, the distributed solutions are scalable and may lead to good delivery performance. However, the content discovery when there is no control over content location is costly, and leads to high network overhead.

Each solution basis has advantages and disadvantages. Based on the solutions surveyed, Figure 2.1 depicts the input data, advantages and disadvantages of each of the three solution basis classes selected. In general, *Index-based* solutions include the following characteristics: scalable since they require only local information; robust to topology changes; and fault-tolerant, because new indices are computed as soon as changes take place. However, these solutions cause a high network overhead due to message exchange between vehicles to help with index computation. *Self-allocation* solutions are also scalable because vehicles make decisions autonomously, and fault-tolerant because many replicas of the same content may coexist. However, it is costly to keep current content. Finally, *Graph-based* algorithms lead to high delivery rates. On the other hand, they are computationally complex and require foreseen and accurate data to operate.

To conclude our discussion on replica allocation solutions, it is important to state that the great majority of solutions found in the literature only performed evaluations on low-scale mobility scenarios. In addition, some of them only presented analytical results. All evaluation results presented by the authors are extremely relevant for validating their proposals. However, we argue that before deploying a content delivery solution to a real VANET, more realistic evaluations must be conducted. In fact, it is also important to note that there is a lack of realistic vehicular mobility scenarios available in the literature. Hence, another challenge when proposing a VANET content delivery solution is to determine the proper evaluation method.

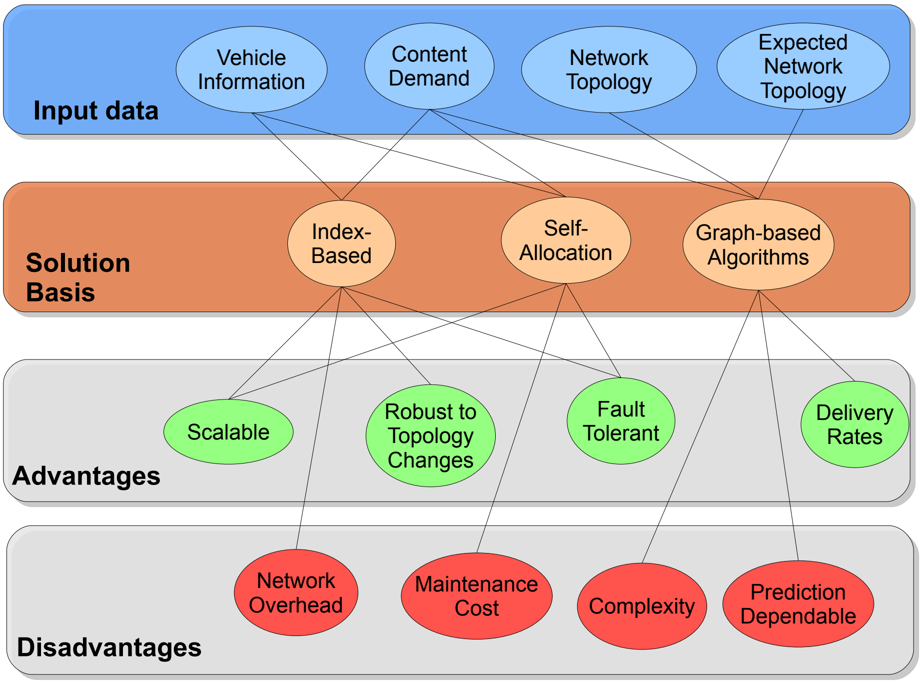


Figure 2.1. Replica allocation solutions general aspects: input data, solution basis and main characteristics.

2.5.3 Content Delivery Solutions

One of the most covered areas in the literature regarding VANETs is routing and forwarding strategies [Willke et al., 2009; Nadeem et al., 2006; Li and Wang, 2007; Bujari, 2012]. Most studies focus on deciding which vehicles should act as relays in the process of forwarding a message to its destination [Rezende et al., 2014; Maia et al., 2013; Villas et al., 2013; Sung and Lee, 2012; Ruiz et al., 2012; Rezende et al., 2012; Rostamzadeh and Gopalakrishnan, 2011; Viriyasitavat et al., 2011; Ciccarese et al., 2009; Zhao and Cao, 2008]. These solutions are important to content delivery as they are responsible for disseminating requests and responses through multi-hop communications. However, they are outside the scope of this thesis since they refer to message forwarding instead of content delivery.

Content delivery in VANETs is classified as pull- or push-based. In a pull-based application, vehicles send requests to the content providers (or their replicas) and receive the requested content from a selected provider. One of the challenges of this approach is the replica discovery process, that decides which replica is most appropriate for responding to a specific request. Unlike pull-based approaches, push-based applications assume that vehicles with specific characteristics are interested in particular content; the objective is to then deliver content to all such vehicles. One of the challenges of this approach is to achieve high coverage where, in the best case, all target vehicles receive the content.

In this section, for each pull- and push-based approaches, we describe and analyze solutions according to their architectural organization as well as to their solution basis. Most solutions adopt distributed infrastructure-based or infrastructure-less strategies, while some of them propose hierarchical solutions for pull-based applications. Some of the studies presented were already described from the replica allocation perspective in Section 2.5.2. Hereafter, they are analyzed from the content delivery point-of-view.

2.5.3.1 Pull-based Solutions

One architectural approach adopted for content delivery is the hierarchical one. In this strategy, the network is organized into clusters, and the cluster heads are responsible for receiving requests from their members and for determine the appropriate provider to whom they can send out a response. In [Gerla et al., 2014b], the authors propose that vehicles in the vicinity elect one of them to be connected to the Long Term Evolution (LTE) network and to then share content with others. Since LTE technology involves a high cost, incentives to users and the round-robin scheme are adopted, so the cost would be shared among the involved vehicles.

Some approaches adopt hierarchical structures for content discovery and delivery in MANET scenarios. Due to the specific characteristics of VANETs, hierarchical solutions have not been applied extensively in these networks. The hierarchical approach is advantageous in terms of scalability issues. It should be noted that the cost of organizing and maintaining clusters may not be feasible in such dynamic networks with constant topology changes, such as VANETs. However, the ideas behind MANET solutions may be of interest to future solutions for VANETs. Hence, some of these solutions are analyzed.

In DHTR [Martin and Hassanein, 2005], the cluster heads use a global replica cache to propagate a request to only those clusters that are supposed to keep the requested content, as opposed to all of them. This solution diminishes the overhead in the network when the content is not located on the same cluster as the requesting node. In [Derhab and Badache, 2006] the authors also propose a hierarchical pull-based content distribution approach in which the requests are sent to the cluster heads. A cluster head tracks the content each of its cluster members are able to provide; based on this information, it decides which member can respond to a request. Nodes send update messages to their cluster heads informing them when their content has changed.

Another common approach to pull-based content delivery is to broadcast requests in multi-hop communications until they reach a provider. The provider then sends the content to the requesting node using the multi-hop reverse path in the direction of the requester. The main problem with one such approach is the flooding scheme used to propagate the request. Thus, efficient forwarding strategies that deal with the broadcast storm problem must be adopted to avoid a high overhead. In InfoShare [Fiore et al., 2005], a content request is broadcast in a flooding scheme until it reaches a source vehicle that can respond to it. Upon receiving the request, the carrier vehicle delivers the content by a unicast path to the requesting vehicle. The path between the content source and its destination is built during the requesting process; this includes the addresses of the relay nodes until the request reaches the carrier. Similarly, CRoWN [Amadeo et al., 2012] proposes new layers to the IEEE 802.11p protocol stack which are responsible for providing content-centric communication. Content requests are broadcast by the consumer until they reach a provider that can respond. To avoid broadcast storms, relay nodes adopt a contention time and only forward a request if they have not received the same request from one of their neighbors.

The broadcasting request approach is also used in CCVN [Amadeo et al., 2013], a pull-name-based content-centric architecture in which vehicles broadcast requests to RSUs and nearby vehicles. Since more than one provider may exist, the most responsive one is selected to deliver the requested content following the reverse path to the requester. Finally, SPAWN [Nandan et al., 2005] adopts a P2P approach similar to traditional swarming protocols like BitTorrent [Cohen, 2003]. However, unlike the traditional centralized approach, peer discovery is done in a distributed way through the broadcasting of gossip messages.

One problem encountered through the propagation of request messages in multi-hop communications is the generated overhead. Additionally, the reverse path to the requesting node may change as a result of the vehicle's movement, affecting the route of the response to the requester. To avoid such overhead costs, vehicles may inform their neighbors about their available contents. Interested nodes will then be able to send requests directly to one of the providers. In [Guidec and Maheo, 2007], mobile hosts announce to their neighbors a list of documents they keep stored locally. When a mobile node is interested in a document, it sends a request to the document owner that, in the sequence, broadcasts the requested document content. According to the authors, broadcasting is used in the response because more than one host may be interested in the same document. Similarly, a file sharing solution is proposed in [Lu et al., 2011], which exploits opportunistic communications between mobile nodes to deliver files to interested nodes.

Some proposals make use of global information from infrastructure stations to schedule a delivery. In this case, request messages are sent to infrastructure stations that are aware of the expected vehicle trajectories, and are able to schedule from the points where the requesting vehicle must receive parts of the content along their route. In the work described in [Malandrino et al., 2013], the authors perform evaluations on the content delivery in VANETs for such a system model. In this scenario, each vehicle may be interested in different content and may send a request to an AP that will schedule the delivery. These authors propose a time varying graph model and, based on realistic scenarios, discover important information, such as how the AP locations play a major role in the network capacity, and knowing how the user mobility can be advantageous in the application of the carry-and-forward communication paradigm.

In MobTorrent [Chen and Chan, 2009], vehicles send their content requests to infrastructure stations, using either cellular communication or other WWAN methods. Based on the request and the expected mobility behavior represented by a predicted contact graph, the station performs a pre-fetch of the content and schedules the delivery by selecting the APs and the vehicles to assist, using opportunistic communications. The requester keep the APs up-to-date by sending information on which chunks have already been delivered.

Network coding is a scheme proposed to improve the overall network capacity [Gkantsidis and Rodriguez, 2005]. Basically, the idea behind this mechanism is that packet forwarders combine several packets together before transmission. Given the broadcast characteristic of wireless links, network coding can help to increase the network throughput [Katti et al., 2008]. VANETCODE [Ahmed and Kanhere, 2006] is a content distribution solution that takes advantage of network coding to help with peer selection and content discovery. Vehicles request content from an AP that encodes the content blocks and broadcasts them to all passing vehicles, including those that are requesting. When not under AP coverage, vehicles cooperate with one another by sending out the blocks they own. Unlike SPAWN [Nandan et al., 2005], vehicles do not have to request specific blocks, because the encoding scheme adopted makes all blocks relevant to vehicles.

Figaro [Malandrino et al., 2012] keeps content location information in the so-called brokers, located in infrastructure stations. Requests are sent to brokers that search for and indicate mobile nodes that could provide the content to the requesting node. Mobile nodes also keep the brokers informed of the content they are able to provide. A broker disseminates the request to other brokers through a proxy when it does not have an entry for requested content. In TEG-PW [Malandrino et al., 2014], when a vehicle wants to download specific content, it sends a request to the query management server via an RSU or via cellular communications. The request is forwarded to RSUs in the area near the vehicle. The RSUs fetch portions of the content from the content server and deliver them to the vehicle. When appropriate, RSUs exploit V2V communication by selecting other vehicles as relays or to carry-and-forward the content.

In [Trullols-Cruces et al., 2012], vehicles send request messages to an AP that uses its contact map to organize a cooperative download scheme. The scheme is achieved by delivering the content to intermediate vehicles that have a high probability of encountering the requesting vehicle. Another similar proposal is CarTorrent [Lee et al., 2007], in which vehicles send their requests to APs that deliver the chunks available for the connection period. In addition, vehicles periodically generate gossip messages to inform others about their content; a V2V communication is then established among vehicles to co-operatively download the remaining chunks.

Some of the pull-based solutions found in the literature consider only infrastructure communication and do not take advantage of V2V opportunistic transmissions. The main drawback of these solutions is infrastructure overload. This is one drawback with MoPADS [Ha and Ngo, 2009], which considers the integration of vehicular and cellular networks. Vehicles send their content requests using the cellular network. Then, MoPADS schedules the content delivery by selecting which APs will be part of the process. To cope with such a NP-hard problem, the authors propose a heuristic to select the delivering AP and determine the content to be delivered, with a focus on maximizing the throughput. This solution also takes into account the expected vehicle trajectory.

The work presented in [He et al., 2013] focuses on video delivery to vehicles in a scenario covered by both cellular and WiFi networks. Since WiFi networks are less expensive and provide higher bandwidth than cellular ones, the proposed algorithm uses this technology as its first option. The on-road video delivery problem tackled in the article was proven to be NP-complete, and the authors proposed heuristics to solve it. The APs enrolled in each delivery, and the period of time and location that the cellular communication will adopt, is obtained by the server that schedules the delivery. One difference in this work related to similar ones in the literature is the requirement that the video experience have good quality, which adds restricted delay constraints to the problem.

To conclude the pull-based content delivery analysis, we describe a solution in which the content is pulled by base stations from vehicles to collect their information. In DMND [Wang et al., 2010], the base stations are the interested parties of the model, and request information from vehicles using a named data approach. They periodically broadcast messages containing the content name that interests them to nearby vehicles. Upon receiving a request message, a vehicle decides if it must respond based on the information supplied by request message naming.

In the pull-based applications, content providers respond to specific requests to deliver the requested content. When infrastructure stations are available, the main challenge relates to the delivery scheduling process; this selects the vehicles and stations to act as providers, depending on their contact prevision with the requester. On the other hand, the content discovery process is the main challenge when no infrastructure stations are available. In this case, it is important to use efficient routing schemes and to select good replica keepers to perform content pre-fetching.

2.5.3.2 Push-based Solutions

In push-based applications [Willke et al., 2009], vehicles presenting particular properties are assumed to be interested in content, and must thus receive it. One strategy adopted is to allow RSUs to periodically broadcast their content to passer-by vehicles. The vehicles that receive the content may then act as disseminators to help with the propagating process. DP [Zhao et al., 2007] is a data dissemination scheme in which a data center selects specific roads on which to push the content, based on a disseminating zone. The content is propagated to the selected roads and vehicles passing by, which then use a broadcast contention scheme to disseminate the data to the desired dissemination zone. In addition, some vehicles passing by selected intersections that may lead to the dissemination zone are also selected to carry-and-forward the content. In InfoCast [Sardari et al., 2009], it is considered that all vehicles in a highway scenario are interested in all messages originating from the RSUs. The RSUs adopt a rate-less coding scheme and broadcast their messages to all vehicles passing by. When a vehicle receives a message in its entirety, it is considered to be a carrier and thus broadcasts the message to its neighbors to help increase coverage. In [Baiocchi and Cuomo, 2013], the list of content available in the server-side stations is pushed to vehicles passing by, which disseminate this information to other vehicles using V2V communication. Vehicles interested in content use cellular communication to request and download it. In other words, the push-approach is used to inform vehicles about the availability of content. Type-Based Content Distribution (TBCD) [Cao et al., 2014] adopts a similar approach. The content is first pushed by a provider to the RSUs located close to the interested vehicles. Then, the RSUs periodically broadcast the content to the passing vehicles. In addition, some vehicles also rebroadcast the content, depending on the content type and on the number of interested clients.

When the content is already located in the vehicles, they may propagate this information to their neighbors using V2V communication. *PrefCast* [Lin et al., 2012] is a solution that considers a mobile social scenario in which nodes forward their content to their neighbors using opportunistic communications. The forwarding is performed considering user profile and preferences instead of relying on the proximity of the mobile nodes. Although proposed for MANETs, the idea of social preferences may be applied to VANETs to improve content distribution solutions. RTAD [Sanguesa et al., 2015] is a real-time adaptive dissemination system in which vehicles decide the broadcast scheme to use, among a set of schemes, based on the current network density and topology information on the road. Thus, this solution is expected to perform well under different network conditions. Push-and-Track [Whitbeck et al., 2012] is a framework that takes advantage of opportunistic ad-hoc communications to offload the network core infrastructure when disseminating content to various nodes. This solution was proposed for scenarios in which many nodes may be interested in the same content. Some nodes are selected to initially receive the content, which they then periodically disseminate to their neighbors. One strength of this solution is that the disseminators keep track of the nodes that have already received the content by adopting an acknowledgment scheme, which is very useful for increasing the dissemination coverage area.

In some scenarios, content should be pushed to a single vehicle. The TSF (Trajectory-based Statistical Forwarding) [Jeong et al., 2010] solution uses the target vehicle trajectory to send the message to an RSU (target point); this RSU will in turn become the rendezvous point for the vehicle (i.e., an RSU that the vehicle is expected to encounter). RSUs are selected in the vehicle trajectory based on the time they are expected to pass the RSUs, and the expected delay. An encounter prediction map is used in [Xu et al., 2011] to schedule content delivery from source to destination. In this solution, APs are responsible for collecting and offering trajectory information pertaining to vehicles. Based on an encounter prediction graph, the delivery of a message is scheduled using multi-hop V2V communications. Existing push-based dissemination approaches take into account the direction and movement of vehicles to decide which vehicles to send a message to, as is proposed in [Nadeem et al., 2006]. In this study, vehicles disseminate traffic information to other vehicles depending on their movement direction.

An alternative to the pure push- and pull-based dissemination approaches is the adoption of the Publish/Subscribe (P/S) [Eugster et al., 2003] paradigm; in this paradigm, content is delivered only to subscribers that have shown interest in particular content. This approach was explored in [Leontiadis et al., 2009a; Leontiadis and Mascolo, 2007; Leontiadis et al., 2010, 2009b]. Vehicles that have an interest in content send out a subscription message to express their interest. When content is available, the publishers push content only to those vehicles that have subscribed to it.

The hierarchical architecture is also adopted in push-based dissemination, as proposed by [Derhab and Badache, 2006]; in this work, cluster heads are responsible for periodically propagating new data updates to their cluster components, and also to other cluster heads. Before disseminating updates, the cluster heads wait for a period of time to receive more updates from other nodes, reducing the message exchanges.

In [Maihöfer et al., 2005], the authors evaluate three *geocast* push-based dissemination approaches: server, election, and neighbor. In the first approach, a server sends a message to the destination region. It can then deliver the message either periodically or by notification. Depending on the distance from the server to the destination region, this solution may not perform well. In the election approach, a node in the destination region is elected to store and disseminate messages. Although efficient in terms of coverage, the election approach generates a high overhead in the network. Finally, in the neighbor approach, each node keeps the messages destined for its location, and shares them with a new neighbor in the destination region. As a drawback, we can mention the bandwidth required to deliver a message in the destination region.

2.5.3.3 Remarks

The decision of using a pull- or push-based approach depends on the application demands. If content must be delivered only to requesting vehicles, the pull-based approach is more appropriate. Otherwise, if the content providers must decide which vehicles need to receive content according to their properties, the push-based approach is more appropriate. However, both pull and push-based solutions may differ depending on the technique used to deliver the content.

Each solution may have different types of data as input. We classify the input data in five categories: *Network Topology*, *Expected Network Topology*, *Vehicle Information*, *Content Information*, and *Network Information*. The definitions of the first three (i.e., *Network Topology*, *Expected Network Topology*, *Vehicle Information*) are straightforward and are the same as described in Section 2.5.2.4. *Content Information* refers to content demand as well as content meta-data such as type and author. Finally, *Network Information* is used by some solutions as network measurements, for example, link quality, congestion, and capacity.

In addition to the different types of input data, each content delivery solution can also be classified according to its solution basis. In this thesis, we classify them in four categories: *Reverse Request Path*, *Delivery Scheduling*, *Periodic Broadcast*, and *Content Announcements*. In the *Reverse Request Path* solutions, request messages are disseminated until they reach a vehicle that can respond; in other words, a provider.

The provider then responds to the requester using the reverse path (i.e., the path in the opposite direction) that the request message took to reach it. On the other hand, *Delivery Scheduling* solutions usually use the expected network topology to schedule a delivery based on the expected contacts that the requester will have in the future. In contrast, in *Periodic Broadcast* solutions, providers periodically broadcast content to vehicles passing by. Finally, *Content Announcements* refers to solutions in which content providers announce their content to their neighbors. A vehicle interested in certain content then requests it directly from a known provider.

The main solutions found and described in this thesis are summarized in Table 2.4, which presents their input data, solution basis, and some comments. Each content delivery solution technique has its advantages and disadvantages as well. Figure 2.2 highlights each solution input, advantages, and disadvantages. Application designers should refer to these results before deciding on the best approach for their particular demands.

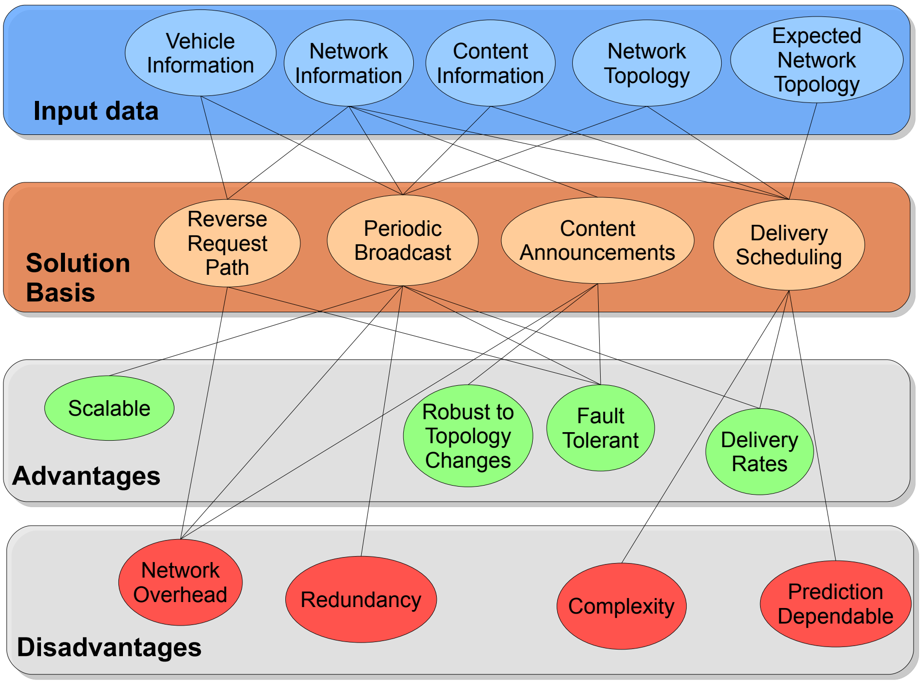


Figure 2.2. Content delivery solutions general aspects: input data, solution basis and main characteristics.

Reverse Request Path is one of the most frequently adopted techniques for content discovery and delivery, particularly in distributed architectures. In this method, the path that a request message travels in order to reach the provider is used following the reverse direction for the delivery of content. The main drawbacks of such an approach

Solution	**Input**	**Solution Basis**	**Comments**
Pull-Based			
LTE Driven Cluster [Gerla et al., 2014b]	Network Information, Vehicle Information	Reverse Request Path	Clustering management
InfoShare [Fiore et al., 2005]	Network Information	Reverse Request Path	Overhead caused by requests
CroWN [Amadeo et al., 2012]	Network Information	Reverse Request Path	Overhead caused by requests
CCVN [Amadeo et al., 2013]	Network Information	Reverse Request Path	Overhead caused by request
SPAWN [Nandan et al., 2005]	Network Information	Content Announcements	Overhead caused by replica discovering
MobTorrent [Chen and Chan, 2009]	Expected Network Topology	Delivery Scheduling	Depends on prediction accuracy
VANETCODE [Ahmed and Kanhere, 2006]	Network Information	Periodic Broadcast	Overhead caused by periodic messages
Figaro [Malandrino et al., 2012]	Network Information, Content Information	Delivery Scheduling	Overhead caused by advertisements messages
TEG-PW [Malandrino et al., 2014]	Expected Network Topology	Delivery Scheduling	Depends on prediction accuracy
CarTorrent [Lee et al., 2007]	Network Information	Content Announcements	Overhead caused by gossip messages
MoPADS [Ha and Ngo, 2009]	Network Information	Delivery Scheduling	Depends on prediction accuracy
OVD [He et al., 2013]	Vehicle Trajectory	Delivery Scheduling	High computational complexity
DMND [Wang et al., 2010]	Content Information	Periodic Broadcast	Overhead caused by periodic messages
Push-Based			
DP [Zhao et al., 2007]	Network Information, Vehicle Information	Periodic Broadcast	Overhead caused by periodic messages
Infocast [Sardari et al., 2009]	Content Information	Periodic Broadcast	Overhead caused by periodic messages
PrefCast [Lin et al., 2012]	Content Information	Delivery Scheduling	User profile requirement
RTAD [Sanguesa et al., 2015]	Vehicular Information	Periodic Broadcast	Overhead caused by beacon messages
Push-and-Track [Whitbeck et al., 2012]	Network Topology, Vehicular Information	Periodic Broadcast	Cost to track all vehicles covered
TBCD [Cao et al., 2014]	Vehicle Information	Periodic Broadcast	Content information requirement
TSF [Jeong et al., 2010]	Vehicle Trajectory	Delivery Scheduling	Overhead caused by beacon messages
STDFS [Xu et al., 2011]	Expected Network Topology	Delivery Scheduling	Overhead caused by beacon messages

Table 2.4. Content Delivery Solutions Summary

are the overhead caused by flooding requests in large-scale networks, such as VANETs, and the fact that the reverse path to the requester may not be the same due to high vehicle mobility. Thus, an efficient routing protocol that deals with the broadcast storm problem must be adopted. In addition, the replica allocation scheme plays an important role in its performance: when a replica is found quickly, fewer messages are exchanged, and fewer hops are required. On the other hand, this technique is scalable since only localized information is required; it is also fault-tolerant, since more than one path to the provider may exist.

Another scalable and fault-tolerant technique that leads to high delivery rates is the *Periodic Broadcast*. However, this solution also leads to a high number of redundant messages and, consequently, a high network overhead. In addition, good replica allocations will also impact positively on its performance. In *Content Announcements* solutions, content providers announce to their neighbors the list of content they are able to offer. This also leads to a high network overhead because of the announcement messages. However, the content discovery process is less complex and less expensive.

One option is to adopt a hybrid solution that supports distributed protocols and takes advantage of infrastructure information. However, the *Delivery Scheduling* solutions require a significant amount of information concerning vehicles' movements, traffic conditions, road maps, traffic lights, etc. In addition, the adoption of such information most likely requires the execution of complex graph algorithms.

2.6 Final Remarks

In this chapter, we presented an in-depth survey of the literature in terms of content delivery concepts applied to VANETs. All studies and solutions found and discussed reinforce the idea that CDN and P2P concepts seem to help the design of VANET applications. In addition, it is clear from these studies that application designers have no models nor frameworks to follow in order to increase the chance of success when developing and deploying their applications. Also, the studies lack detailed and large-scale evaluation and are proposed to address particular problems. Therefore, in the next chapters of this thesis we fill this gap by proposing a hybrid model called Vehicular Content Delivery Network (VCDN), and a framework to help application designers. We also propose and evaluate, under large-scale scenarios, two solutions that, in addition to validate our VCDN model, also advance the state-of-the-art in this field as they are novel in terms of how to replicate content in dynamic VANETs.

Chapter 3

Vehicular Content Delivery Networks

Vehicular ad hoc network (VANET) applications are evolving from simple alert message exchanging to advanced systems with content delivery demand [Costa-Montenegro et al., 2012]. Typically, these emerging applications require heterogeneous content to be delivered to vehicles, including traffic notifications, weather reports, advertisement videos and images, entertainment videos, and other sort of files to be shared with vehicles and their passengers. However, the task of delivering content in such a dynamic environment is easier said than done [Gerla et al., 2014b]. Therefore, substantial research efforts in this area is expected to bring such applications into reality.

Usually, two approaches have been used to deliver content in the traditional Internet [Passarella, 2012]: Content Delivery Network (CDN) and Peer-to-Peer (P2P). CDN solutions rely on the replication of content in the so-called surrogate servers strategically placed in the network, and on redirecting a request to the server most able to respond to it. As result, CDN provides high content availability in an infrastructure-based approach. However, it assumes the existence of stationary strategically located servers to replicate content. On the other hand, nodes in P2P solutions cooperate among themselves by offering their resources to their peers, leading to a scalable solution. Usually, nodes can join and leave a P2P network whenever desired, which is a good strategy in terms of scalability, fault-tolerance, and deployment issues. However, content discovery and delivery in P2P networks may take a long time and generate a high network overhead.

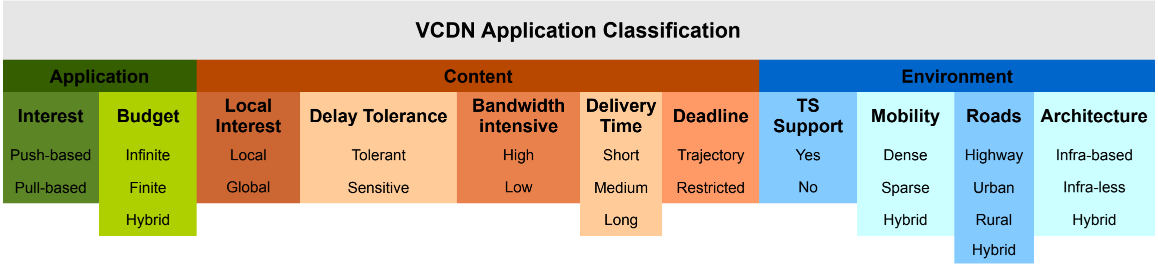
Several characteristics of VANETs and their applications suggest that pure CDN and P2P models, as originally conceived for the Internet, are not suitable for them. First, many applications in VANETs are referred to as push-based, meaning that content should be pushed to the clients even in the absence of a request, like an accident notification content, for example. In addition, contacts in VANETs are intermittent, making the establishment and maintenance of end-to-end links very difficult. Furthermore, the deployment of a surrogate server in a large-scale urban scenario is a costly and time-consuming task. When it comes to content, several entities are potential sources, such as vehicles, Intelligent Transportation System (ITS) solutions, Wireless Sensor Networks (WSNs), mobile users, among others. Moreover, content sources and clients move in considerable speed, causing constant changes in the network topology. Finally, content in many applications is location- and time-dependent, meaning it is valid only inside a region of interest (RoI) during a given period.

In this thesis, we advocate the need for a new model called Vehicular Content Delivery Network (VCDN), where aspects of both CDN and P2P are integrated, adapted, and extended to VANETs. On one hand, the infrastructure formed by stationary surrogate servers, as in CDNs, is used to improve content availability. In this case, we extend the traditional CDN concept by replicating content and storing it also into moving vehicles. On the other hand, the ad hoc and self-organizing nature of P2P networks is exploited, and vehicles cooperate among themselves to discover and deliver content. We also adapt these services to VANETs by considering the vehicular mobility. In summary, our proposed model exploits the advantages of each approach, CDN and P2P, and adapt them to the highly dynamic VANET scenarios. As result, our VCDN model is scalable, fault-tolerant, mobility-aware, and works even in the absence of infrastructure stations.

The design of a VCDN requires many decisions to be taken and most of them are directly related to the application's characteristics. In the literature, some studies have already proposed classification schemes for VANET applications, such as [Karagiannis et al., 2011; Bai et al., 2012; Lee et al., 2014; Zeadally et al., 2012]. However, they mainly focus on the user benefits of the applications and on the network aspects, such as critical latency and routing, which are not enough to help designers of content delivery applications. In Section 3.1, we expand the existing classification schemes by proposing content-oriented classification criteria composed of key aspects to the implementation of content replication, discovery, and delivery tasks.

VANETs are expected to run on a variety of environments having different entities and infrastructures. In this scenario, the application's designer should decide about its internal components. To help on this task, we propose, in Section 3.2, our VCDN model and a framework containing the modules to be implemented in order to make content available to the potential clients.

In Section 3.3, we present examples of scenarios to illustrate the flexibility of our



VCDN Application Classification

Application		Content					Environment			
Interest	**Budget**	**Local Interest**	**Delay Tolerance**	**Bandwidth intensive**	**Delivery Time**	**Deadline**	**TS Support**	**Mobility**	**Roads**	**Architecture**
Push-based	Infinite	Local	Tolerant	High	Short	Trajectory	Yes	Dense	Highway	Infra-based
Pull-based	Finite	Global	Sensitive	Low	Medium	Restricted	No	Sparse	Urban	Infra-less
	Hybrid				Long			Hybrid	Rural	Hybrid
									Hybrid	

Figure 3.1. A list of key aspects that should be considered when classifying a VCDN application. The application aspects should be classified prior to its design and deployment to increase its chance of success.

model and framework that can be adopted by different applications, as well as by a multi-application domain.

In summary, in this chapter we contribute to the research community by proposing a novel content delivery model conceived specifically to VANETs. In addition, we also define a framework that can be used to implement such model. Finally, we present a classification scheme to help application designers on making important decisions on implementation and deployment details.

3.1 VANET Applications Classification

Emerging VCDN applications differ among themselves in several key aspects that affect their design and development. Therefore, before making decisions about design and implementation details, it is recommended to classify the application being proposed in terms of those aspects.

Existing classification schemes [Karagiannis et al., 2011; Bai et al., 2012; Lee et al., 2014; Zeadally et al., 2012] lack key aspects when used to analyze content delivery applications. Thus, we expand these schemes by proposing content-oriented criteria to help designers of content delivery applications, as illustrated in Figure 3.1. We organize the criteria into three major categories: application, content, and environment. The application category comprises criteria of the application as a whole. For a particular application, the content category describes each of its content to be delivered. Finally, the physical scenario where the application is expected to run is described in the environment category. The criteria for each category are described as follows.

3.1.1 Application Category

Content Interest: applications expected to push content to vehicles, considering some particular properties such as vehicle is inside a region of interest (RoI) or moving towards a RoI, or when the users' profile match the content profile, are referred to as *Push-Based*, since content is pushed to the target vehicles regardless of explicit requesting. On the other hand, *Pull-Based* applications require vehicles to explicitly request the specific content they are interested in. Content providers respond to them upon receiving request messages.

Budget Constraints: content delivery to vehicles involves communication costs. Moreover, vehicle-to-infrastructure (V2I) communication is usually more expensive than vehicle-to-vehicle (V2V) transmissions, since it involves base stations, as discussed below. Thus, applications must be aware of how much they are willing to spend on delivering content to their clients. Hence, they can be classified into three classes. The ideal application presents an *Infinite Budget* when cost is not a concern at all. On the other hand, other applications have a *Finite Budget* that must not be exceeded during their execution. Finally, an alternative approach is to have an infinite budget for V2V communication and a limited budget for V2I transmissions only, which we classify as *Hybrid Budget.*

3.1.2 Content Category

Local Interest: Content may be of interest only to vehicles that are inside a RoI, such as traffic information and particular advertisements. We refer to them as *Local* content applications. Differently, content referred to as *Global* is assumed to be of interest to all vehicles across the network scenario, like a city-wide advertisement.

Delay Tolerance: Content that does not tolerate delay (e.g., real-time videos) are classified as *Delay-sensitive.* For such content, delays affect the users' quality of experience. Conversely, *Delay-tolerant* content is not affected by delays, such as an ordinary file to be downloaded and consumed when ready.

Bandwidth intensive: Content may require an intense use of bandwidth, and then is classified as *High* bandwidth intensive. Examples include a large content to be received in a short period, such as streaming of real-time videos. On the other hand, *Low* bandwidth intensive content is properly delivered even under low transmission rates.

Delivery Time: The delivery time refers to the amount of communication time required for a content to be totally delivered to a client. Thus, this metric depends on the content size and on the adopted communication technology. We classify the content delivery time as *Short*, *Medium*, or *Long* when the time required to transmit it is less than one second, more than one second and less than five seconds, or more than five seconds, respectively. These values were defined based on the 802.11p transmission rate that ranges from 3 Mbps to 27 Mbps [Li, 2012], allowing the transmission of up to 3.3 MBytes per second and 16.8 MBytes in five seconds. Larger content requires longer contact duration with a provider in order to be fully delivered, while small content may be fully delivered within a short contact.

Deadline: Even clients of delay-tolerant content may require the delivery to be completed within a particular deadline. This deadline is classified as *Restrict* when content is supposed to be consumed on-the-way, within a specific deadline. In other words, content should be delivered and consumed by the vehicle before its arrival at its destination. Examples include a text content to be read or a non-real-time video to be watched by users before arriving at their destination. When the deadline is the expected arrival time, we categorize it as *Trajectory-based*, meaning that content will be consumed after arrival. For example, a user may download a file during his/her journey to be consumed upon arriving at work.

3.1.3 Environment Category

Transportation System (TS) Support: When public transportation vehicles (e.g., bus, subway) or even taxis are available to support the application, we say it is *TS-supported*. Otherwise, the application is said to be *Non-TS-supported*. Examples of such application include a geo-localized notification to be delivered in a particular bus trajectory.

Mobility Scenario: Vehicular mobility patterns change significantly over time and space. Traffic volume may be extremely large during peak hours, while not so intense in another time, for example. Thus, applications have to be prepared to operate under *Dense*, *Sparse*, or *Hybrid* density scenarios.

Road Scenario: Different road configurations require different solutions in vehicular networks. Hence, applications may have to be able to operate at *Highway*, *Urban*, *Rural*, or *Hybrid* scenarios.

System Architecture: The infrastructure available also plays an important role on the application design. When only infrastructure stations are available to be exploited through Road Side Units (RSUs) or Cellular coverage, the application is considered *Infrastructure-based* and takes advantage of V2I communication only. In this case, V2V is not an option. Conversely, *Infrastructure-less* applications exploit only V2V communication. Finally, a *Hybrid* scenario is an alternative when both V2I and V2V communications are available.

The VCDN application classification is the first contribution of this chapter. This proposed classification extends previous schemes since it considers not only users benefits and network aspects, but also other key features that affect the design of content delivery applications. The classification of an application is a key step that should be done prior to its design, since decisions on later steps depend on how the application is expected to behave. After classifying the application, a designer should then decide its internal structure and distributed components. To this end, in the following, we describe our proposed VCDN model and a framework that encompasses the fundamental building blocks to guide the application design and implementation.

3.2 Vehicular Content Delivery Network

In this section, we present arguments to endorse our proposal of a hybrid model that inherits concepts from both CDN and P2P networks. The objective here is to describe and present the weakness of the pure CDN and P2P models when applied to VANETs. We then propose the integration of both approaches and discuss how this novel model would lead to a better content delivery solution. In addition, we propose a flexible framework to help application designers to implement and deploy applications following our hybrid VCDN model.

3.2.1 Pure CDN Model

The pure CDN approach requires a first step of placing physical servers, in which surrogate servers are strategically placed in geographical regions where potential clients are expected to be. In the VANET context, the surrogate servers must be placed on the roads as Road Side Units (RSUs), so vehicles would be able to wirelessly contact them to request and receive content. After that, content must be placed in the appropriate surrogate servers to be as close as possible to their potential clients. These servers form an infrastructured network used for content sharing and searching services. In pull-based applications, a client vehicle sends a request message to the surrogate server closer to it that will proceed to deliver the locally available content, or to search for it in other surrogate servers. Differently, in push-based applications the surrogate servers are responsible for delivering content to target vehicles even in the absence of request messages.

The main advantage of this pure CDN approach, considering the VANET context, is the infra-structured network formed by the surrogate servers, leading to high content availability. A good alternative for content sharing and searching is to take advantage of high-speed connections between the surrogate servers. However, a high cost, in terms of money and time, is required to deploy and maintain the surrogate servers considering the outdoor, large-scale scenario of VANETs. Furthermore, this approach is not fault-tolerant since the failure of a surrogate server may cause disconnections and, consequently, uncovered regions. Finally, only vehicle-to-infrastructure (V2I) communication would be used, which is more expensive than the ad hoc vehicle-to-vehicle (V2V) communication.

3.2.2 Pure P2P Model

A pure P2P model may also be adopted. The unstructured P2P approach is more appropriate to VANET scenarios mainly because of their highly dynamic topology that makes it difficult to organize and maintain distributed structures. In this case, vehicles joining the network establish a connectivity with their neighbors in an ad hoc manner, forming a self-organizing and decentralized network. Content in such a pure P2P approach is shared among vehicles that collaborate to increase their availability. To motivate collaboration, vehicles that cooperate with others receive benefits, such as priority in packet routing, when downloading content. In pull-based applications, a client vehicle disseminates a request message that propagates through the network until reaching providers that send the requested content back to the client. Differently, in push-based applications, providers (i.e., vehicles keeping content) deliver content to the vehicles potentially interested in it, regardless of the existence of request messages.

The main advantage of such pure P2P approach is its fault-tolerance, since the failure of a provider is not perceived given the existence of other servers nearby. In addition, there is no deployment cost and the self-organizing nature of P2P network fits well in the dynamic scenario of VANETs. Nevertheless, there is a high cost in searching for content in terms of delay and network overhead, given the highly dynamic topologies posed by VANETs and the poor content availability that may occur. In addition, when a provider is found, the client may have moved to a different location, increasing the delivery cost.

3.2.3 Proposed Hybrid Model

Our hybrid model takes advantage of the benefits from both CDN and P2P approaches, and extends some of their concepts to VANETs. On one hand, we exploit the idea of replicating content on surrogate servers running on RSUs to increase content availability. Furthermore, we extend such concept by allowing content to be replicated also in moving vehicles as well, which increases content availability and fault-tolerance. On the other hand, P2P concepts for distributed content discovery and delivery are exploited, together with incentive mechanisms that benefit cooperative peers. To reduce the overhead and the time to find a content in the network, our model takes advantage of the infrastructured stations to track the content availability nearby and then, to indicate potential providers to clients. To help on the content replication and delivery tasks, our model includes a mobility management service that monitors mobility patterns and supports the decision of content replication, discovery, and delivery.

Our model is fault-tolerant because a failure in a surrogate server, either a RSU or vehicle, is compensated by others nearby. Moreover, a content is expected to be found near all vehicles given its availability in RSUs and in other vehicles, reducing the overhead caused by P2P content discovery protocols. The possibility of vehicles acting as surrogate content providers also makes our model scalable, which is a key factor for large-scale VANETs. In addition, our model works in either infrastructured and infrastructureless scenarios, since the use of RSUs is not mandatory. Finally, our model deals with the highly dynamic topology of VANETs by implementing a mobility management service.

Figure 3.2 illustrates an example of how the main entities of our model interact. In general, our proposed VCDN model has three stages: infrastructure planning and placement, content replication, and content discovery and delivery.

Infrastructure planning and placement: This stage is a pre-application step that involves placing infrastructure stations (i.e., RSUs) in appropriate locations. Given the high cost involved in deploying the infrastructure, engineers and urban planners, who will find the appropriate locations to optimize the network coverage, should head this task. It is important to note that this stage should be executed considering the requirements of a variety of future potential applications.

Content replication: An application replicates its content to RSUs and vehicles

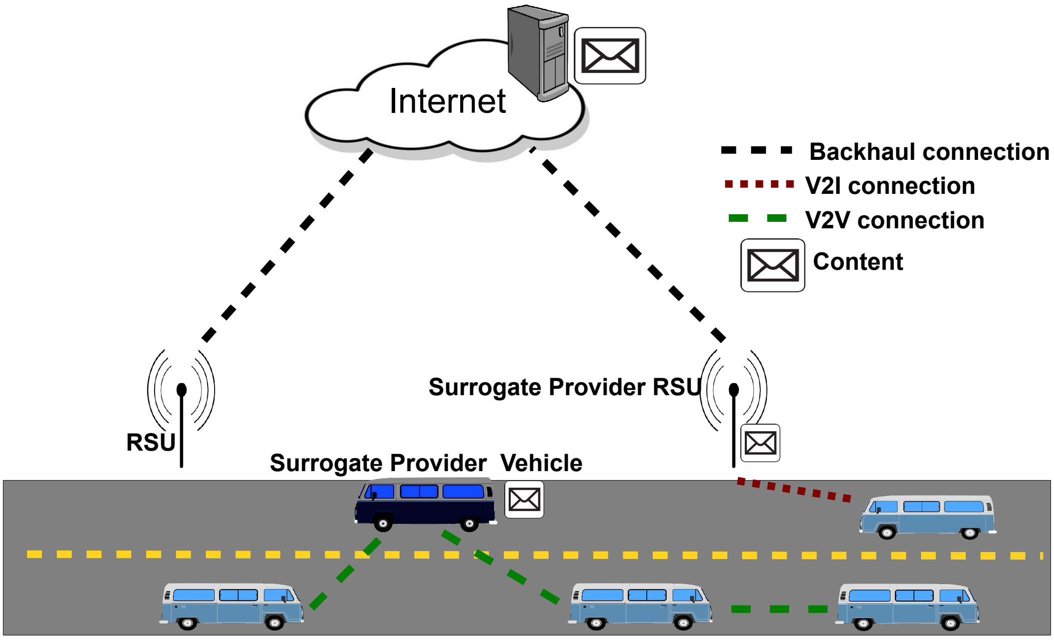


Figure 3.2. General system architecture of our proposed hybrid model. In this example, one RSU and one vehicle are selected as surrogate content providers, keeping replicas of content. The other vehicles are covered either directly by the surrogate RSU (using V2I communication) or by the surrogate vehicle (using V2V communication). The other RSU acts as content tracker, keeping information about the content available in its vicinity. The other vehicles also act as relays in the multi-hop V2V communication. It is important to note that our model is also compliant with infrastructure-less architectures, not illustrated in this example.

to make it available to potential clients The decision of when and where to replicate depends fundamentally on the application's and content's characteristics, such as local interest, delay-tolerance, expected traffic conditions, content size, among others. In addition, the expected mobility pattern is also a key factor to decide where to place content replicas, since contacts among vehicles themselves, and vehicles and RSUs, will determine how well a content is delivered to its clients. Vehicles and RSUs selected to replicate content take the role of surrogate content providers, helping in the content delivery task. Content providers are monitored in terms of how useful they have been on content delivery and, whenever necessary, new replicas are allocated and useless ones are removed with the objective of increase content availability.

Content discovery and delivery: In this stage, content is delivered to its clients upon requesting it explicitly (i.e., pull-based applications) or not (i.e., push-based applications). Regarding the push-based applications, the content delivery is straightforward: content providers, either origin or surrogates, deliver the content periodically to potential clients. The list of potential clients depends on the application and may include those that satisfy some properties/characteristics such as current location, trajectory, weather status and current time. To motivate cooperation, surrogate providers acting in push-based applications are monitored and receive benefits in their future request.

In contrast, clients in pull-based applications explicitly discover potential providers and send request messages to them. The content discovery relies on the dissemination of look up messages that are received by content providers that, in turn, respond to the client offering the content (or parts of it) available. In addition to its available content, providers also inform its resources availability depending on the historical cooperative behavior of the client. In other words, a provider may allocate more network resources to more cooperative vehicles to motivate collaboration among vehicles. The client then chooses, among all offers received and based on expected mobility pattern, the most appropriate providers, and sends request messages directly to them requesting the content (or part of it). Upon receiving the requesting messages, the providers send response messages containing the requested content (or part of it) to the client.

It is important to mention that we focus on the application level, assuming the existence of network-level addressing rules and routing protocols. Thus, our proposal is compliant with the Information-Centric Network (ICN) architecture [Ahlgren et al., 2012; Amadeo et al., 2013; Liu et al., 2014], which proposes, among other things, a name-based approach for content delivery and in-networking caching schemes.

3.2.4 Framework

The proposed VCDN model requires important decisions to be made, including the components to be part of it and how those components are supposed to communicate among themselves. The authors of the challenge paper described in [Bai and Krishnamachari, 2010] have already proposed a high-level generic networking framework for content delivery in VANETs. We go further and present a lower-level framework, responsible for making content available to its clients.

Our framework is flexible to be adopted by different applications with different demands. This is a very attractive characteristic, given the diversity of VANET applications that are becoming a reality. In addition, each module may run on different

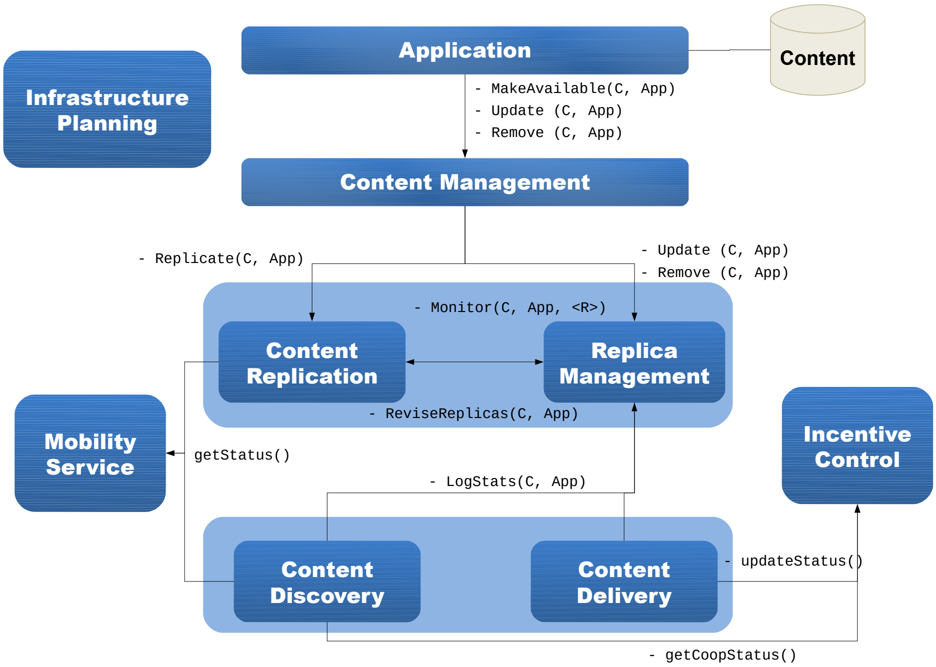


Figure 3.3. Framework encompassing the internal modules a VCDN should implement. The interfaces among the modules help understanding the tasks performed by each of them. It should be noted the application (App) parameter in the module's interface, which makes the framework compliant with multi-application domains.

entities, depending on the application.

The framework's modules are illustrated in Figure 3.3 and described in the following.

Infrastructure Planning: This module represents the pre-application step of planning and placing of RSUs. A typical approach for such task is to adopt optimization algorithms to select where to place RSUs to obtain a high coverage with a low cost. It is important to mention that this module should consider the execution of different applications in the future, and, therefore, should be well planned and implemented. The output of this module is a list of locations where RSUs should be placed. When the infrastructure is already deployed, or an infrastructure-less scenario is expected, this module can be ignored.

Mobility Service: When it comes to VANETs, the vehicular mobility plays a determining role on decision making. The *Mobility Service* module is responsible for providing mobility status regarding particular areas and vehicles. Mobility status, in this case, refers to the vehicles' trajectory, vehicles' expected temporal contact graph, expected network density of a particular region, and vehicles' points of origin and destination. The implementation of this module may vary from scenario to scenario, and can be integrated with Intelligent Transportation System (ITS) entities, such as traffic monitoring tools and semaphores, to improve the accuracy of the mobility status.

Content Management: This module provides services to the applications to make content available to its potential clients, as well as to update or remove existing content. Upon beginning its execution, the application provides key characteristics to this module, such as content size, lifetime, target vehicles, RoI, delay constraints, budget, among other aspects according to the classification scheme proposed in Section 3.1. The *Content Management* module then proceeds to increase content availability by invoking the replication module. Whenever the application demands new content to be delivered, or changes in existing contents, this module is invoked. It should be noted by the module's interfaces that the proposed framework is compliant with multi-applications running simultaneously, since all provided services require the application as a parameter.

Content Replication and Replica Management: The *Content Replication* module is responsible for selecting appropriate vehicles and RSUs to act as surrogate content providers. This complex task is considered as a major challenge given the highly dynamic topology of VANETs. Therefore, the implementation of this module must rely on important aspects such as content size, target vehicles, RoI, delay constraints, budget, and environment characteristics (e.g., available infrastructure and TS-support). Also, this module accesses the *Mobility Service* to obtain the mobility status of involved vehicles and areas, which will play an important role on the replication.

There are many approaches to select the most appropriate replica placements for a content in a VANET, including graph-based, optimization and distributed algorithms, as discussed in Chapter 2. The decision of which one to adopt in a particular case depends on aspects related to the application, the content to be replicated, and the environment. In addition, there is a lot of research opportunities in this field, given that the appropriate replication is a fundamental issue to improve content discovery and delivery tasks. Given the diversity of VANET applications, our framework is flexible to implement any replication solution that is more appropriate according to the designers' decision. Therefore, we do not specify any particular implementation detail.

After replicating the content, the *Content Replication* invokes the *Replica Management* module that is responsible for monitoring the selected replicas. Tracking the replicas is a key functionality to keep replicated content consistent, since changes have to be applied to all replicas as soon as possible, to avoid the delivery of out-of-date content. When content must be updated or removed, this module provides the information of the current replica nodes that will be affected. The modules responsible for discovering and delivering content update the *Replica Management* module with statistics about the performance of the current surrogate providers, in terms of the number of clients, the amount of data provided, and the average throughput of each one of them. Thus, when a surrogate provider is not performing well, the *Content Replication* service is invoked to revise the replication with the objective of increasing the content availability.

Content Discovery and Delivery: The task of delivering depends fundamentally on the application's content interest (i.e., push- or pull-based). For push-based applications, replica nodes (i.e., surrogate providers) should deliver content to all target vehicles, expected to be interested in it. One challenge, in this case, is the estimation of how relevant the content is to the target vehicles. On the other hand, clients in pull-based applications must explicitly request content from the providers. In this case, the *Content Discovery* module is responsible for finding the most appropriate providers to respond to a request. To find a provider, a client disseminates a message looking up for a content. Initially, the look up message is disseminated with TTL (time-to-live) of one hop. If no providers answer within a configured period, the same look up message is disseminated with TTL of two, and so on. This procedure is repeated until a maximum configurable value for TTL is reached, or providers respond offering the content. It is important to mention that the replication process is an important step to reduce the time of finding a content.

Available surrogate providers that receive the look up message respond with a message containing the available parts of the content they have, together with their mobility status and the network capacity. Among all potential providers and their mobility status, the client selects the most appropriated to request the content. To reduce the time of finding appropriate providers, the RSUs also act as content trackers by keeping up-to-date information about the vehicles acting as surrogate providers in their vicinity. To this end, vehicles make use of beacon messages to inform the RSUs in their range about the content they have available. Therefore, tracker RSUs also respond to look up messages informing which vehicles could provide a particular content.

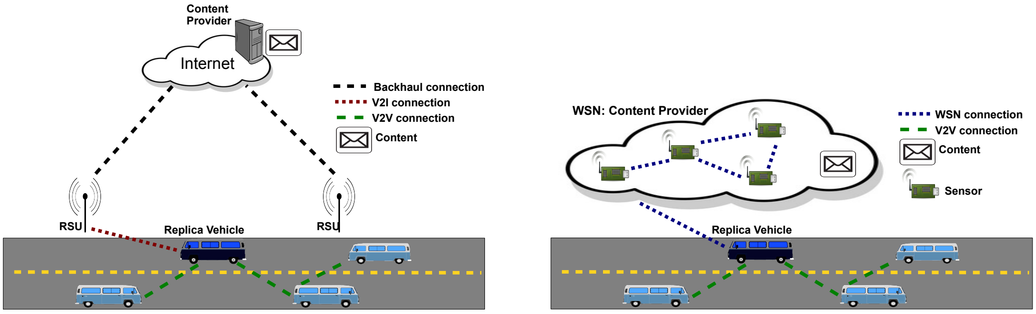
The *Content Delivery* module is, therefore, responsible for delivering content from providers to clients. Supposing a pull-based application, the providers discovered in the *Content Discovery* service deliver the content to the client either directly or using multi-hop communication. In the latter, relay vehicles are exploited so the message reaches its destination. On the other hand, push-based applications adopt a periodical broadcast approach that can be configured in terms of the delivery periodicity and the coverage distance (i.e., number of hops).

Incentive Control: VCDN applications also require constant cooperation from vehicles to operate properly. For example, vehicles must offer their resources to act as surrogate providers in exchange for some benefits. In addition, facilitator vehicles will play an important role on multi-hop communication to act as packet forwarders. Therefore, the *Incentive Control* module is responsible for defining and controlling how vehicles will benefit from cooperating with each other. To this end, the *Content Delivery* service informs the amount of data delivered and relayed by vehicles, which is used by the *Content Discovery* service when providers offer their content to clients. In other words, providers should offer their resources according to the clients' cooperative behavior. However, to give new opportunities to selfish vehicles to cooperate, content providers offer their resources randomly to vehicles with such characteristics. In this way, they will receive the content and may then start cooperating. Again, our framework is compliant with different incentive mechanisms, such as credit-based and reputation-based schemes.

The proposed VCDN model and framework are expected to help application designers to model the system from the logical point-of-view. The aforementioned framework is flexible enough to be implemented on different entities, forming different scenarios depending on the available infrastructure. In the next section, we present potential scenarios for the implementation of a VCDN that will illustrate the use of our framework.

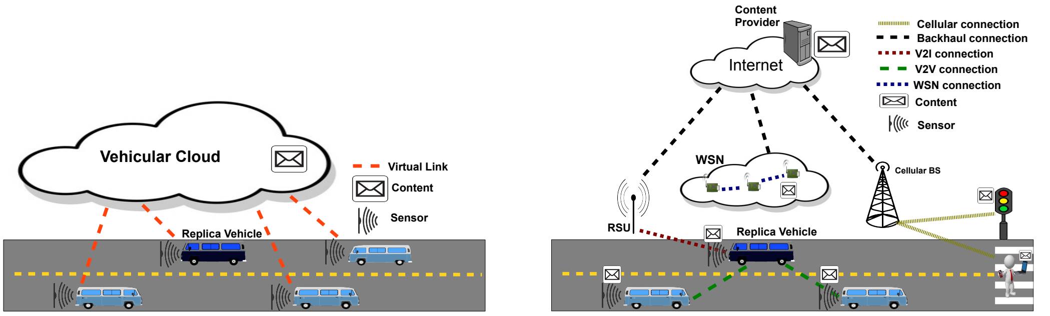
3.3 VCDN Application Scenarios

The proposed VCDN model and framework are flexible enough to be adopted in a variety of scenarios, assuming the logical modules are implemented according to their definition. In this way, the logical modules of the framework can be implemented on different physical entities. Figure 3.4 depicts four possible organizational architectures that might be adopted. In the following, we describe potential applications for each one of those scenarios. It should be mentioned that the objective here is to show how the VCDN model and framework should be adopted in different scenarios. Therefore, we do not present details regarding the implementation of the applications described.



(a) Internet Server as content provider.

(b) WSN as content provider.



(c) Vehicles organized in a vehicular cloud as content providers.

(d) VANETs as part of a Smart City, with different content providers.

Figure 3.4. Examples of application architectures for a VCDN.

3.3.1 Image file shared from the Internet

Figure 3.4(a) illustrates an infrastructure-based architecture where content is provided by an Internet server. The objective of this application is to make a large image file of an advertisement available to the interested vehicles inside a region of interest (RoI). The content file, which is divided into several fragments, should be pushed to the vehicles in the RoI, assuming a hybrid budget where the V2V communication is preferred. Also, an urban and dense scenario is expected, given the application should run for an entire day, facing periods of heavy as well as low traffic. No TS-support is available, and the server uses RSUs to reach the vehicles.

To achieve its goal, the server itself runs the *Content Management*, *Content Replication*, and *Replica Management* modules that together select and monitor the replica vehicles to keep the content available inside the RoI. The number of simultaneously allocated replicas is dynamic and based on the current and expected density of the RoI, which is inferred from beacon messages sent periodically by the vehicles. Regarding the *Content Delivery* module, the replica vehicles are responsible for delivering, periodically, the image file to their neighbors in the RoI. The Internet server, responsible for monitoring the overall process, gives incentives to vehicles. Thus, vehicles that act as replicas or facilitators (i.e., packet forwarders) receive the benefits of having priorities when downloading another file from the server.

3.3.2 WSN for traffic monitoring

In the infrastructure-less scenario shown in Figure 3.4(b), a Wireless Sensor Network (WSN) monitors the traffic by counting the number of vehicles in the roads, and makes such information available to vehicles. The objective of this application is to provide traffic information to vehicles. In this case, content is pulled by vehicles interested in the traffic information from the WSN and from replica vehicles. Also, content is small, delay-tolerant, with restricted deadline, and local to a medium-size area of relevance to vehicles in it. The application should work in a hybrid density scenario, where traffic density may range from sparse to dense in the occurrence of an accident, for example.

The *Content Management*, *Content Replication*, and *Replica Management* modules run on vehicles executing decentralized algorithms. Vehicles decide about the replicas using comparable indices indicating each vehicle's mobility pattern in terms of speed, direction, and distance from the monitored region. The index value for a vehicle indicates how good the vehicle is likely to keep the content available to others. The *Content Discovery* and *Delivery* modules run the basic look up, request and response solution proposed by our VCDN model. Vehicles that act as replicas receive, by the *Incentive Control* module, a priority when requesting a traffic information in their route.

3.3.3 Vehicles for traffic monitoring

Another pure distributed, infrastructure-less scenario is depicted in Figure 3.4(c), where vehicles gather data from their own sensors and make it available to others. The objective of this application is to provide traffic information inferred by the vehicles, without an external support. In such case, vehicles are virtually part of a vehicular cloud network responsible for sharing content and resources whenever possible. Vehicular cloud is a recent topic discussed in the literature [Lee et al., 2014].

Vehicles in the cloud monitor their mobility behavior in terms of speed and acceleration, and infer the occurrence of a traffic jam based on the roads' characteristics. When a traffic jam is inferred, this information is made available to others in the cloud. Content in this case is a small notification message valid for a short period in a specific region where the traffic jam is taking place. The *Content Management*, *Content Replication*, and *Replica Management* run in the vehicular cloud, transparently for the vehicles. The *Content Discovery* and *Delivery* modules, as well as the *Incentive Control*, also run in the cloud and rely on the basic solution of our VCDN model. The application is expected to work in a hybrid density scenario, both in urban and highway roads.

3.3.4 Smart City

Finally, Figure 3.4(d) illustrates an envisioned Smart City scenario where VANETs are part of an overall Intelligent Transportation System (ITS). In such scenario, ITS equipments, such as smart semaphores and traffic monitoring systems, as well as pedestrians using smart mobile devices, act as content providers and clients. Vehicles are also equipped with sensors to collect and make content available to others. In addition, a WSN is also part of the overall ITS system. With respect to the Internet connections, cellular network coverage is available additionally to the RSUs, which increases the infrastructured network capacity. Several applications are expected to run simultaneously in such scenario, which is a requirement accomplished by our VCDN framework.

As an example, suppose two running applications: traffic monitoring and entertainment system. In the traffic monitoring, information from different sources is aggregated to improve the current and future traffic situation. Vehicles monitor their mobility status, a WSN monitors the number of vehicles and their speed passing through it, and the smart semaphores equipped with cameras provide information about their schedule and the traffic movement around them. All that data is aggregated in an Internet server that processes it and makes the traffic information available to vehicles, as well as to online services. Since the server has information about the vehicles' position and trajectory, it uses graph algorithms to select appropriate replica vehicles for each content. Vehicles interested in traffic information request it from the Internet server using the RSUs or the cellular network. The Internet server orchestrates the delivery by selecting the replica vehicles based on their trajectory and expected network topology.

The entertainment system, on the other hand, is responsible for providing entertainment videos to on-board users. Videos are large and have delay-sensitive content expected to be pulled by users and fully consumed on-board. They are provided by an Internet server that is also responsible for replicating them close to potential clients according to their personal profile collected from online social networks. Users with mobile smart devices participate in this application by providing opinions about videos in social networks. Given the delay-sensitive content characteristic, the replication should be aware of the performance of the delivery, to increase the number of replicas when the quality-of-experience faced by the clients is not as high as expected.

As stated earlier, the VCDN model and framework proposed in this work are fully compliant with such multi-application domain. The designer must, however, pick the most appropriate entities to run each of the framework's modules, depending on the objectives and characteristics of each application, according to the criterion list discussed in Section 3.1.

3.4 Final Remarks

In this chapter, we advocate that CDN and P2P systems, as originally conceived for the Internet, are not appropriate to VANET applications. Therefore, we propose a hybrid model in which concepts from both approaches, CDN and P2P, are exploited and adapted to VANETs. To facilitate the adoption of our model, we also propose a framework that encompasses the fundamental building blocks of our VCDN model. Finally, we discuss how our model could be applied to different scenarios. The next step is to adopt the proposed model in real VANET applications with the objective of validating its benefits. However, before that, we present in the next chapter vehicular mobility characterization results to give insights on how to implement the VCDN building blocks.

Chapter 4

Vehicular Mobility Characterization

A key aspect that turns the problem of content delivery particularly challenging for VANETs is the vehicular mobility. The network topology changes as vehicles move, making the contacts between them significantly intermittent. This is an important issue since end-to-end connections may not last for the entire period required for a content delivery. Thus, having insights on how vehicles move may bring benefits when selecting appropriate surrogate providers and transmitting content. Given that, in this chapter we present characterization results of a realistic, large-scale vehicular mobility trace. First, we present graph-based characteristics that indicate that some vehicles are more likely to be better content providers than others. Then, we characterize the vehicles in terms of their origin-destination (O-D) points, which are easily obtained and can be useful in content delivery solutions, as it will be showed in the next chapter.

4.1 The Mobility Data Trace

The data trace used in this thesis consists of a publicly available vehicular mobility trace from Cologne, the fourth-largest city of Germany [Uppoor et al., 2014a]. Its metropolitan area encompasses over 400 km^2 and more than one million inhabitants. Cologne has nine regions, which can be organized in two main groups: downtown and suburb. The downtown, called *Innenstadt*, is the most populous and dense traffic region. The remaining eight regions are comprised of suburbs.

The trace contains trips (i.e., a vehicle's movement from its departure position until its arrival position) that occurred between 6:00 am to 8:00 am on a weekday. The trace comprises trips from more than 120,000 vehicles departing from and arriving at different places throughout the city. The original trace was improved and validated by Uppoor et al. [Uppoor et al., 2014b; Uppoor and Fiore, 2011], who have applied several techniques and corrections to improve its accuracy. As far as we are concerned, this trace is the most complete, realistic, and large-scale vehicular mobility data available in the literature. Furthermore, Cologne's characteristics are similar to many cities around the world, which makes it a good model to be characterized, since the results may be replicated in other scenarios.

4.2 Graph-based Characterization

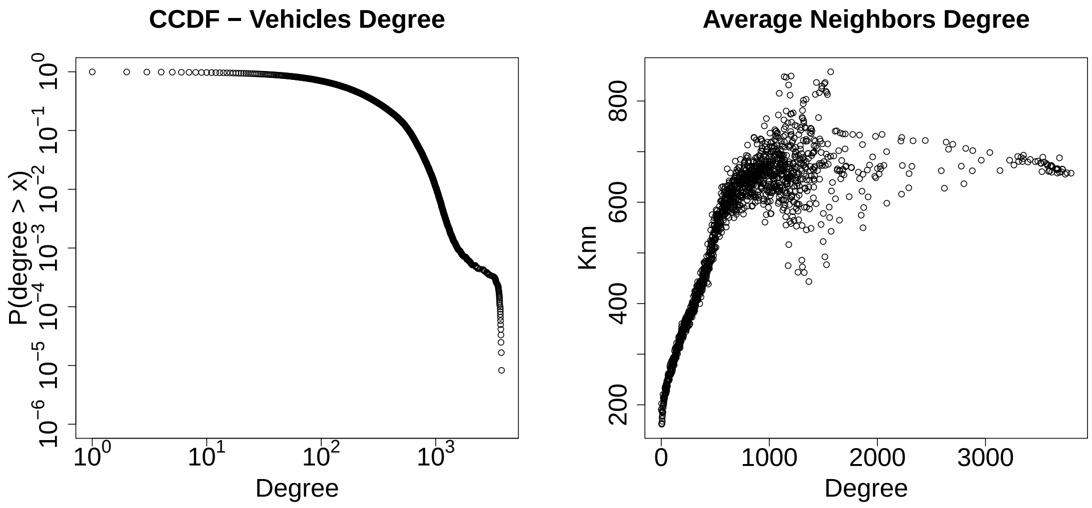
Our proposed VCDN model seems to be very attractive to VANET applications. However, one question that arises is whether or not some vehicles are indeed more appropriate than others to be the replica placements. In other words, would it be worth to spend computational and network resources to select content replicas in such scenario?

Some studies in the literature tried to answer the question of how vehicles differ in their mobility properties. In [Resta and Santi, 2010], the authors demonstrated analytically that the fully capacity of data dissemination in mobile networks can be achieved only when the best disseminating nodes are selected. In the study described in [Zyba et al., 2011], the authors demonstrated for different mobility scenarios (taxis and mobile users in a University Campus) that some mobile nodes can be considered more relevant to data dissemination than others; they furthermore demonstrated that the dissemination capacity is expected to improve when those nodes are selected.

Also, some authors have already studied the behavior of mobility traces to discover helpful insight. Xia et al. [2012] characterized a mobility trace composed of 12,096 taxis from Beijing during a week, focusing on contact and clustering characteristics. Ahmed and Salil [2010] analyzed 1,200 buses from Seattle in regarding to contact duration, inter-contact intervals, and clustering characteristics. In [Monteiro et al., 2012], simplified urban (4Km^2) and highway (25Km) scenarios were also characterized. To the best of our knowledge, the only characterization of realistic urban scenario was performed by Uppoor and Fiore [2012], where the authors analyzed the trace from Cologne, in Germany, in regarding to vehicles flow and network density.

Nevertheless, other aspects of the network should also be studied to help on answering the posed question. Thus, we go further and characterize the realistic, large-scale vehicular mobility trace from Cologne to show that some vehicles present special characteristics regarding complex networks and mobility aspects.

To assess the characterization results, the network was modelled as a graph $G = (V, E)$ in which V represents the set of vehicles and $E \subseteq V \times V$ is the set of edges (e.g. contacts among vehicles). An edge $e = (v_i, v_j)$ between vehicles v_i and v_j exists if the



(a) Vehicles degree distribution

(b) Knn (Pearson Coefficient: 0,69)

Figure 4.1. Vehicles degree distribution follows a power-law on its tail. There is a positive correlation between degree and k_{nn} values

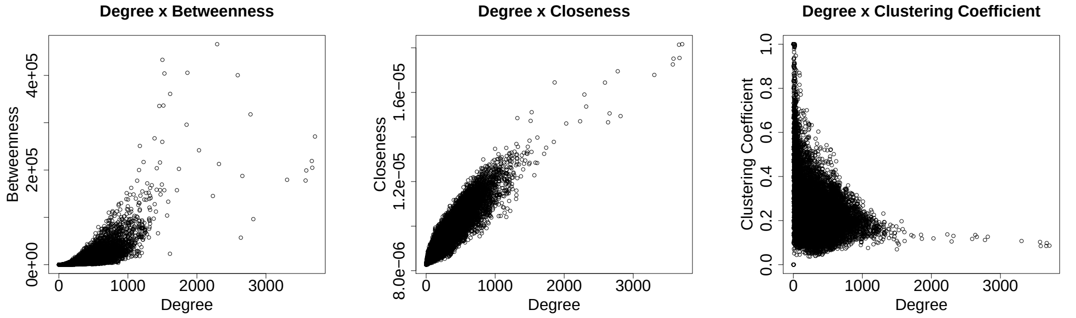
distance $d(v_i, v_j)$ between v_i and v_j was lower than 100 meters for a period of time. The value of 100 meters was chosen based on experimental results from [Cheng et al., 2007; Teixeira et al., 2014].

In the following we present the characterization results in terms of complex networks and mobility aspects.

Degree distribution The degree distribution of vertices is an interesting metric that provides relevant information on the network topology. We here consider the complete graph comprised of all vehicles and their contacts during the entire trace period. This graph comprises 120,913 vehicles and 15,385,919 contacts among them.

Figure 4.1(a) presents the vehicles degree complementary cumulative distribution function (CCDF) in log-log scale. Most vehicles have a small number of contacts while a small fraction of them have significantly more contacts (minimum=1, median=356 and maximum=3,792). The Maximum Likelihood Estimator (MLE) method was used to check whether this distribution follows a power-law or not, like in other social networks [Mislove et al., 2007]. Considering all values ($xmin = 1$), the high value of the Kolmogorov-Smirnov statistic (KS = 0.64) indicates that this distribution does not follow a power-law. On the other hand, considering only the tail ($xmin = 798$), we can say that it follows a power-law distribution for $\alpha = 5.88$ (KS = 0.034 in this case).

Besides the degree distribution, we have also measured the K_{nn} metric that maps



(a) Degree x Betweenness (Pearson Coefficient: 0.74)

(b) Degree x Closeness (Pearson Coefficient: 0.92)

(c) Degree x Clustering Coefficient (Pearson Coefficient: -0.30)

Figure 4.2. Centrality and clustering relationships with vertices degree

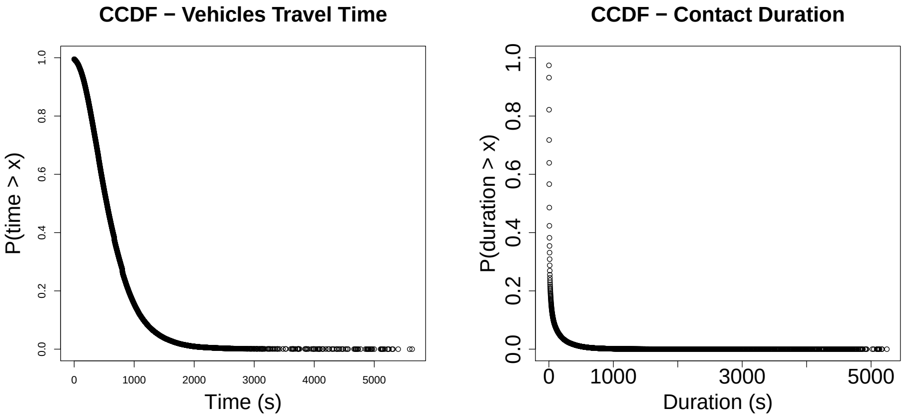
vehicles' degree to their neighbors average degree. As showed in Figure 4.1(b), the positive correlation between vehicles' degree and K_{nn} values (Pearson coefficient = 0.69) indicates a tendency of vehicles with higher degree also to be in contact with others with this characteristic.

Centrality and clustering Centrality and clustering metrics also give insights regarding the network topology. Figure 4.2(a) depicts the relationship between vehicles' degree and their correspondent betweenness values. The betweenness of a vertex v_i is the number of minimum paths between any two other vertices that include v_i. The higher the betweenness, the more important to the network the vertex is. We can note a correspondence between the vertex degree and its betweenness. However, there are some vehicles with higher betweenness that do not present higher degree.

The closeness is also a centrality metric calculated by the inverse of the sum of all distances between a vertex v_i and all other vertices. The higher this value, the lower is the sum of the distances and therefore the vertex is relatively close to the others. Figure 4.2(b) shows that there is a linear correspondence between vertices' degree and closeness since its Pearson coefficient is 0.92.

Another important metric is the clustering coefficient that measures the edges density of a vertex's neighbors. As depicted in Figure 4.2(c), vertices with lower degree have a higher clustering coefficient in general.

Travel time It is also important to characterize vehicles movement and the contacts among them. Figure 4.3(a) presents the CCDF of the travel time vehicles take to reach



(a) Travel time for vehicles

(b) Contacts duration

Figure 4.3. Movement and contact information. Some vehicles travel for longer periods than the majority ones. Furthermore, some contacts among them lasted for many minutes while most contacts last less than 15 seconds.

their destination; it is possible to note that there are a small number of vehicles that travel for significant longer periods than the majority. We have also studied how long each contact among vehicles lasted. Figure 4.3(b) presents the CCDF of the length of time that each contact lasted and it allows us the realize that most contacts lasted for few seconds, while some of them lasted for much longer periods. This metric is relevant since the higher the contact time, the more data can be transmitted.

Based on the characterization results, it is possible to come up with some insights that validate the idea that some vehicles are more likely to be better content replicas than others:

- some vehicles present a significantly higher number of contacts than most others;
- centrality analyzes show that some vehicles present higher values of *betweenness* and *closeness*, indicating they are more important to the network structure than others;
- clustering coefficient analysis indicates that some vehicles are more connected to their neighbors than others;
- a small fraction of vehicles travel for longer periods than most others;
- a small fraction of connections between vehicles last for longer periods of time than most others.

As a result, we conclude that some vehicles have special characteristics that turn them more likely to be better replicas than others, and that characteristics such as degree, clustering coefficient, and travel time are relevant to the identification of such vehicles. However, topology metrics, such as vehicle's degree and clustering coefficient, are difficult to predict before having the entire contact graph. On the other hand, the vehicle's travel time is easily estimated by navigation systems given the origin-destination (O-D) points, specially with the help of online traffic information applications. Therefore, in the next section we characterize the same mobility trace from the O-D perspective to find how knowing the departure and arrival points of vehicles could be useful for content delivery applications.

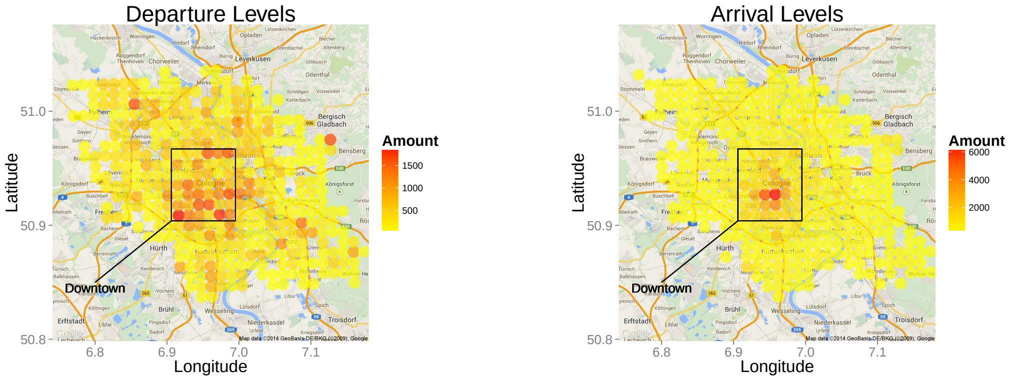
4.3 Origin-Destination-based Characterization

Many existing solutions for content delivery applied to VANETs require either the current or the expected network topology, which are difficult to obtain given the high mobility of vehicles. In this thesis, we come up with the hypothesis that the vehicles' origin-destination (O-D) points could be very useful when selecting appropriate replica placements. Thus, in this section we present a characterization of the Cologne's mobility trace performed under the O-D standpoint.

4.3.1 Results

We start this process by partitioning the geographic area of Cologne into quadratic sub-areas with sides measuring 1000 m, and then counting the number of departures and arrivals for each of those sub-areas. The 1000 m value was chosen to provide a representative number of samples to the analyses. Note that vehicles departing from and arriving at the same region (i.e., intra-region mobility) are considered in both departure and arrival analyses. Figure 4.4 illustrates the levels of departures and arrivals for each sub-area; it shows that the departures are more equally distributed across the city, while arrivals tend to be concentrated on the downtown region demarcated by the squared area in the center of the maps. Given that, in the following characterization we analyze separately the departures for suburban and downtown regions.

The suburb comprises the eight regions of Cologne, excluding the downtown. Figures 4.5(a) and 4.5(b) present the histograms of the number of departures and arrivals per quadratic area, respectively, for the suburban regions. It is possible to see that a large number of sub-areas contribute to none or a small number of vehicles for both departures and arrivals. Table 4.1 presents the statistical summary of this data.



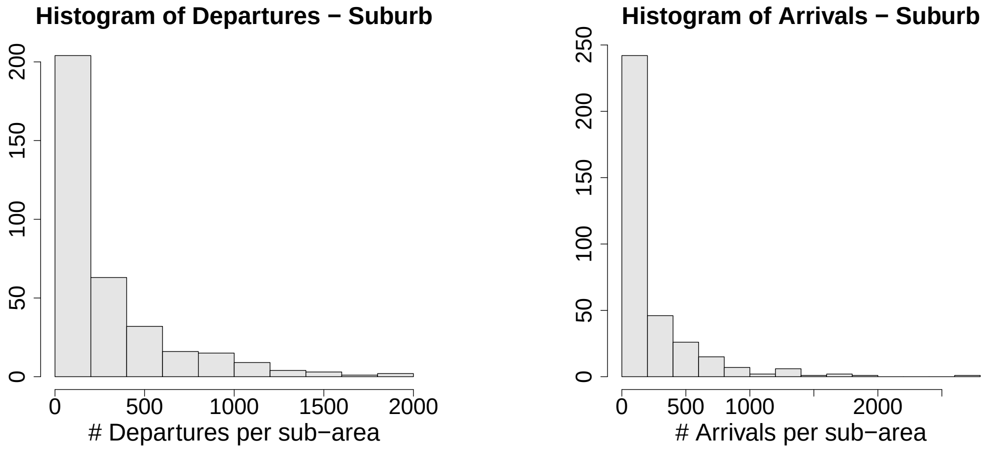
(a) Departures tend to be more equally distributed across the city

(b) Arrivals are concentrated in the downtown region demarcated by the squared area

Figure 4.4. Departure and arrival levels per area

Table 4.1. Suburb statistical summary

Data	Min	Q_1	Median	Mean	Q_3	Max	Var	Skewness
Departures	0.0	3.0	115.0	268.2	387.2	1887.0	132030.0	1.89
Arrivals	0.0	4.0	75.0	210.0	257.0	2698.0	115806.6	3.03



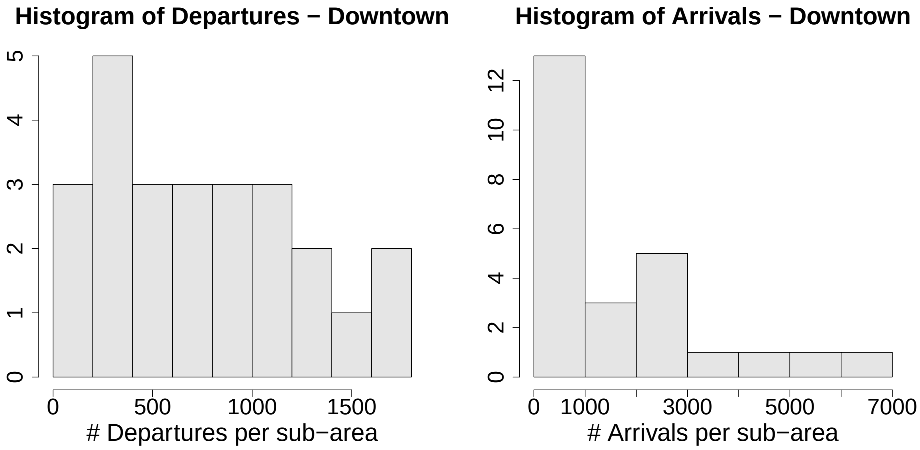
(a) Departures histogram for suburban regions

(b) Arrivals histogram for suburban regions

Figure 4.5. Histograms for suburban departures and arrivals

The same analyses described above were performed for downtown region; the results are presented below. Considering the histograms depicted in Figure 4.6 and the statistical summary found in Table 4.2, it is possible to observe that the downtown region presents a more concentrated flow of vehicles, since its sub-areas have at least

14 departures and 81 arrivals. By the skewness value and the histogram plot, it is reasonable to assume that the downtown departures are represented by a symmetric distribution. Additionally, the arrivals sample presents a high number of small values, and the higher values seem to be uniformly distributed.



(a) Departures histogram

(b) Arrivals histogram

Figure 4.6. Downtown departure and arrival histograms

Table 4.2. Downtown statistical summary

Data	Min	Q_1	Median	Mean	Q_3	Max	Var	Skewness
Departures	14.0	278.0	669.5	714.3	1082.0	1608.0	216630.2	0.28
Arrivals	81.0	659.0	959.0	1684.0	2621.0	6382.0	2783907.0	3.03

The distance traveled by each vehicle and its total travel time are also important metrics. We then analyze the distance traveled by vehicles, as well as the time they take to reach their final destination. In this case, the sample data comprises the distance traveled by each vehicle and its travel time. It is important to state that we consider the departing region of the vehicles in the analyses. In other words, the results referred to downtown/suburb mean that we consider only vehicles departing from downtown/suburb.

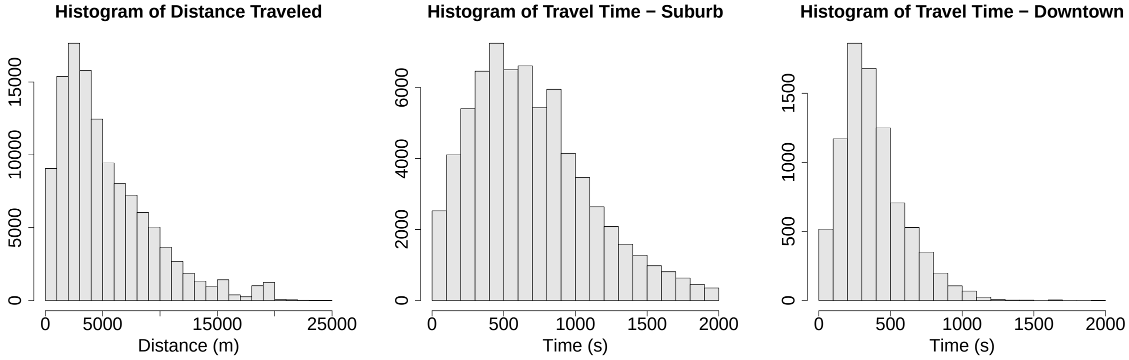
Table 4.3. Distance traveled statistical summary

Data	Min	Q_1	Median	Mean	Q_3	Max	Var	Skewness
All regions	0.0	2327.0	4185.0	5311.0	7395.0	24930.0	16446511.0	1.30

Figure 4.7(a) shows the histogram and Table 4.3 presents the statistical summary of the distance traveled, considering all vehicles departing from both downtown and

suburban areas. For these metrics, analyzing the data separately for vehicles departing from downtown and suburban regions was unnecessary, since they present similar values.

Another important measurement for O-D modeling is the time each vehicle takes to reach its final destination, which may influence the mobility scenario since it impacts the number of vehicles traveling at the same time. Contrary to the distance traveled, the travel time for vehicles differs for downtown and suburban regions. Thus, we present the results separately for these regions. Figures 4.7(b) and 4.7(c) and Table 4.4 present the travel time histograms and the statistical summary, respectively, for vehicles that depart from the suburbs and downtown.



(a) Distance (all regions) (b) Travel time in Suburb (c) Travel time in Downtown

Figure 4.7. Downtown and suburb travel time (in seconds) histograms

Table 4.4. Travel time statistical summary

Data	Min	Q_1	Median	Mean	Q_3	Max	Var	Skewness
Suburb	0.0	380.0	635.0	688.4	929.0	1999.0	166814.7	0.70
Downtown	0.0	225.0	337.0	378.6	487.0	1992.0	47610.46	1.04

In addition to understanding how the number of departures and arrivals are distributed across the city, it is also important to know how those vehicles depart over time. In this analysis, it is only important to evaluate the departure time, as the arrival time depends on the vehicle's destination and on other aspects such as speed and traffic density, which are out of the scope of this characterization.

To perform this analysis, we count the number of departures for each 10-minute time frame. The plot shown in Figure 4.8 presents the empirical Cumulative Distribution Function (CDF) of this data. The mean and standard deviation of the number of departures per time interval are 9816.66 and 2250.20, respectively, and the coefficient

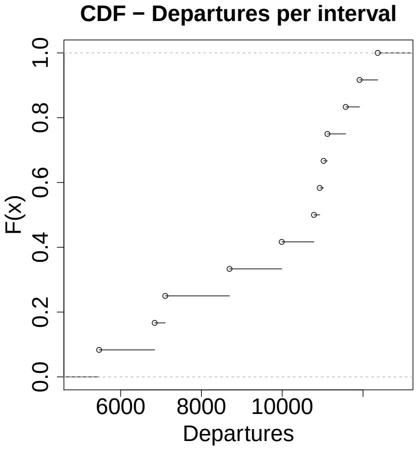


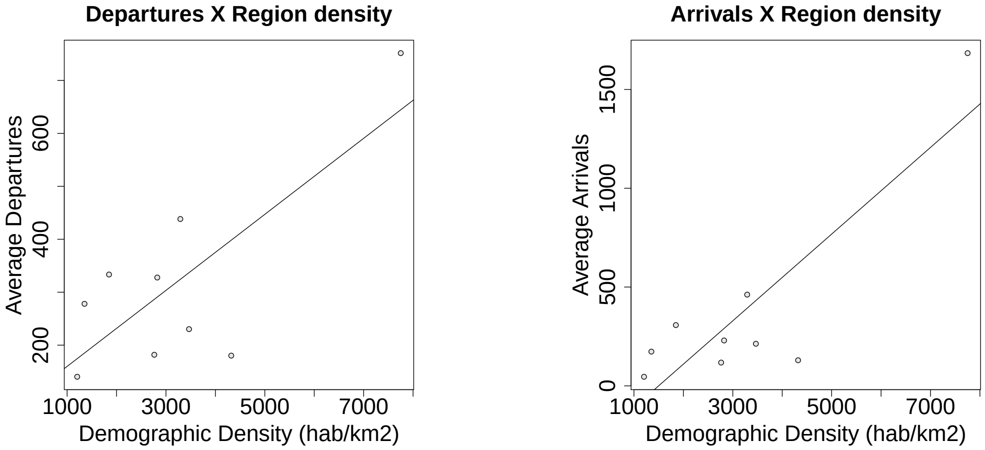
Figure 4.8. CDF of departures for each 10-minute interval

of variation is 0.22. Based on this information and on the plot's visual analysis, it is reasonable to assume that the departures are uniformly distributed over time.

4.3.2 Generic Model

To infer how the aforementioned metrics are related to each city, we argue that the average number of departures and arrivals for a region depend on its demographic density (habitants/area). To confirm our hypothesis, Figure 4.9 presents the linear relationship between the demographic density (x axis) and the average number of departures (y axis in Figure 4.9(a)) and arrivals (y axis in Figure 4.9(b)). It is important to state that the average number of departures and arrivals for a region are calculated by considering all 1000-meter side quadratic sub-areas that comprise the region. We then apply a linear regression model to fit the equations in a form representative of these relationships. The linear equations are in the form $\mu = \beta \times \Delta + \alpha$ where μ is the average number of departures/arrivals for a region, Δ is the demographic density of the region, β is the slope and α is the intercept. By applying a linear regression method, we obtained the values $\beta = 103.10$ and $\alpha = 0.066$ for departures, and $\beta = 0.205$ and $\alpha = -287.05$ for arrivals. We also computed the Pearson and R^2 coefficients to validate the linear relationship between departures/arrivals and the demographic density, as shown in Figure 4.9. Hence, given the demographic density of a region, it is possible to estimate its expected average number of departures and arrivals for each of its 1000-meter sided quadratic sub-areas.

After defining how to compute the average number of departures and arrivals with respect to the demographic density of a region, we apply regression models; these



(a) Departures' linear regression. Pearson Coefficient = 0.76. $R^2 = 0.58$

(b) Arrivals' linear regression. Pearson Coefficient = 0.86. $R^2 = 0.74$

Figure 4.9. Analysis of the relationship between a region demographic density and its average number of departures and arrivals

infer equations for computing the parameter values of the selected probability distributions for all O-D aspects considered. These equations, together with the two defined above, can be used to model mobility data for different scenarios. In the following, the equations for departures and arrivals across the city, departures over time, total travel distance, and total travel time are described.

To this end, the first step is to analyze the statistical data presented above to formulate a hypothesis about the most suitable probability distribution functions $f(X, \theta)$ to describe the data. The goal is to fit the sample data to distributions that are well-known and easy to implement, as the ones presented in Table 4.5. We then adopt the Maximum Likelihood Estimator (MLE) technique [Law and Kelton, 1999; Casella and Berger, 2001] in order to estimate the θ parameter values of the candidate probability functions $f(X, \theta)$ hypothesized in the previous step. The MLE is a well-known and accurate method used for estimating parameters of statistical models that presents interesting properties, such as consistency and efficiency. Finally, all candidate probability functions $f(X, \theta)$ are evaluated in terms of their ability to represent the sample data by using either the Quantile-Quantile Plot (Q-Q plot) or the *Chi-square* (χ^2) goodness-of-fit test [Law and Kelton, 1999; Plackett, 1983] techniques.

Departures across the city The departures per each 1000-meter side quadratic subarea of the suburb are modeled as a Zero Differentiate Geometric distribution with parameters $p = \frac{1}{\mu+1}$ and $pz = \Delta \times 0.15$, where μ is the average number of departures for a region, calculated by the equations defined above, and Δ is the demographic

Table 4.5. Probability distributions

Distribution Name	**Parameters θ**	**Description**
Uniform	$\theta = \{max, min\}$	Each outcome between max and min are equally likely
Geometric	$\theta = \{p\}$	Number of failures before the first success in a sequence of Bernoulli trials with probability p
Negative Binomial	$\theta = \{s, p\}$	Number of failures before the s^{th} success in a sequence of Bernoulli trials with probability p
Zero Differentiate Geometric	$\theta = \{p, pz\}$	Geometric distribution that considers that zeros come from a different process with probability pz
Zero Differentiate Negative Binomial	$\theta = \{s, p, pz\}$	Negative Binomial distribution that considers that zeros come from a different process with probability pz

density of the region. Considering only the downtown region, the departures follow a Uniform distribution with parameters $min = 0.001 \times \Delta$ and $max = 0.20 \times \Delta$, where Δ is the average number of departures for the downtown region. Inside a sub-area, vehicles depart uniformly under the road constraints.

Arrivals across the city The arrivals per each 1000-meter side quadratic sub-area of the suburban and downtown regions are represented as a Negative Binomial distribution with parameters $s = 0.00016 \times \Delta - 0.13300$ and $p = -8.203 \times 10^{-7} \times \Delta + 6.957 \times 10^{-3}$, where Δ is the demographic density of a region.

Departures over time The number of departures over time can be described as a Uniform distribution with parameters $min = 1.70 \times \Delta$ and $max = 3.86 \times \Delta$, where Δ is the demographic density of a region. Within a sub-area, vehicles depart uniformly under the road constraints.

Total travel distance We found out that the total travel distance is well represented by a Negative Binomial distribution with parameters $s = 1.71$ and $p = 0.00032$, regardless of the demographic density of the departure region.

Total travel time This metric can be represented by a Negative Binomial distribution for vehicles departing from downtown and suburban regions. However, the parameters differ, and were inferred as $s = 2.13$ and $p = 0.0030$ for suburban regions and $s = 2.69$ and $p = 0.0070$ for downtown, regardless of the demographic density of the departure region.

4.3.3 Validation

To validate the aforementioned results, we compare them to a data trace available from Zurich, Switzerland [Naumov et al., 2006]. Zurich is the largest city of Switzerland with a population of 1,406,083 people, including its vicinity regions. Zurich's trace was generated based on public census information, and represents vehicles' movement with a high level of realism. It comprises trips of about 260,000 vehicles around an area of 260 km^2 and encompasses 24 hours of traffic movement. However, we consider only the trips between 6:00 am to 8:00 am to be in accordance with Cologne's trace. This interval comprises about 93,000 trips, with vehicles departing from their original position and arriving at their final one. Zurich's trace is adequate for validating since it was generated considering realistic information of Zurich, a city with similar characteristics as Cologne.

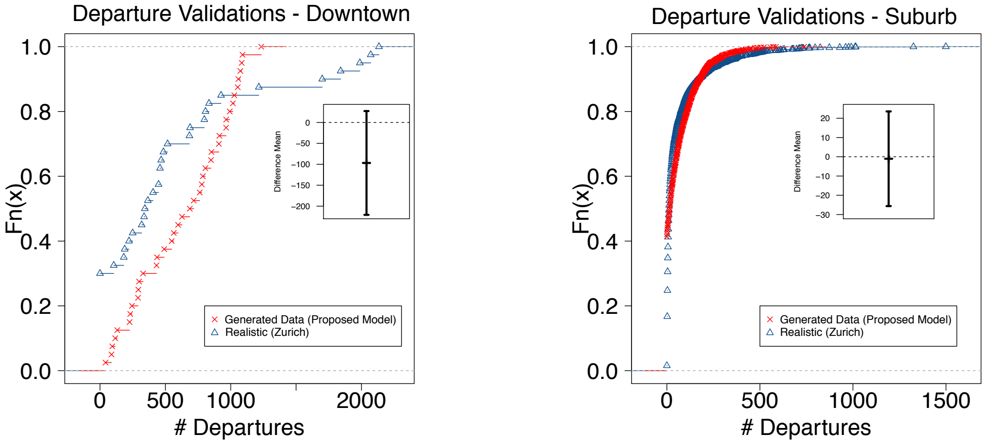
We conduct the validation by generating mobility data, considering Zurich's demographic density as input to our model, and comparing it to the actual data from Zurich's trace. We follow three complimentary approaches to validate our model. In the first one, we adopt the Cumulative Distribution Function (CDF) over-plot procedure, which is one of the heuristic procedures suggested by Law and Kelton [1999]. In this procedure, both generated and actual CDF data are plotted together and if both curves closely agree, it is possible to say that the proposed model was able to generate mobility data similar to the existing ones. To reinforce the over-plot approach, we also calculate the 95% confidence interval of the difference of means of both generated and actual data. To take advantage of the central limit theorem[1], the confidence interval is calculated based on a hundred samples of our model. In this case, if the interval includes zero, both generated and actual data are considered similar. In other words, in 95 out of 100 cases the generated and existing data will be very similar to each other. Finally, we generate a mobility O-D matrix using our proposed model and compare it to the actual matrix from Zurich. This procedure was performed 100 times and the difference of means is plotted in Zurich's map for comparison.

In the validation plots presented in the following, we refer to *Generated Data (Proposed Model)* as the data generated by our model. The actual data is referred to as *Realistic (Zurich)*. The CDF over-plot result is illustrated by the larger, main figure, while the confidence interval result is represented by the small plot.

Figure 4.10 presents the validation results for departure measures in downtown (Figure 4.10(a)) and suburban areas (Figure 4.10(b)). It is possible to see that the generated CDFs for departures in suburban and downtown regions show behavior similar to the actual data. More importantly, both 95% confidence intervals include zero within their lower and upper boundaries. Figures 4.11(a) and 4.11(b) present the expected level of departures for Zurich and the comparison of the generated and actual

---

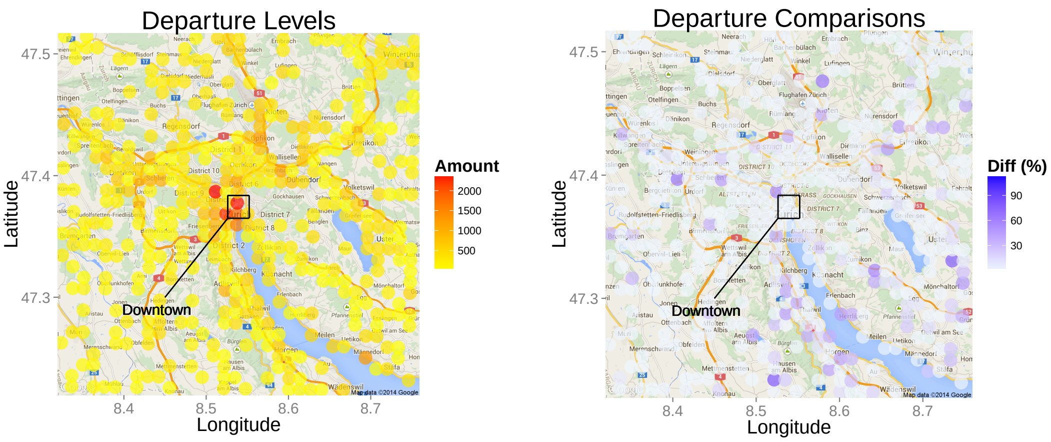
[1] The central limit theorem states that the mean of sufficiently large samples of a population tends to be Normally distributed



(a) Departure CDF Comparison – Downtown

(b) Departure CDF Comparison – Suburb

Figure 4.10. Departure validations.



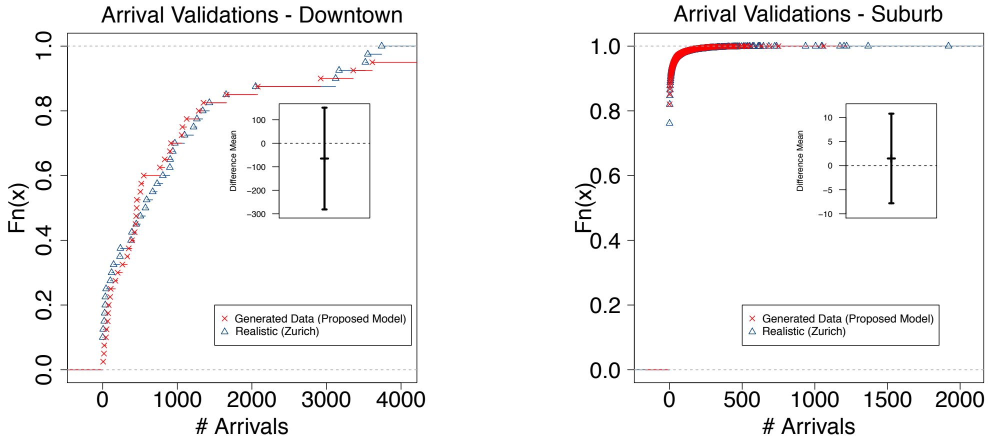
(a) Departure Levels – Zurich

(b) Departure Matrix Comparisons

Figure 4.11. Departure validations.

O-D across the map, respectively. It should be noted that most areas present very similar results, while only a few generated matrices have a difference greater than 30% from the actual matrix. Given this, we can say with reasonable confidence that the proposed model accurately describes the departures of different scenarios.

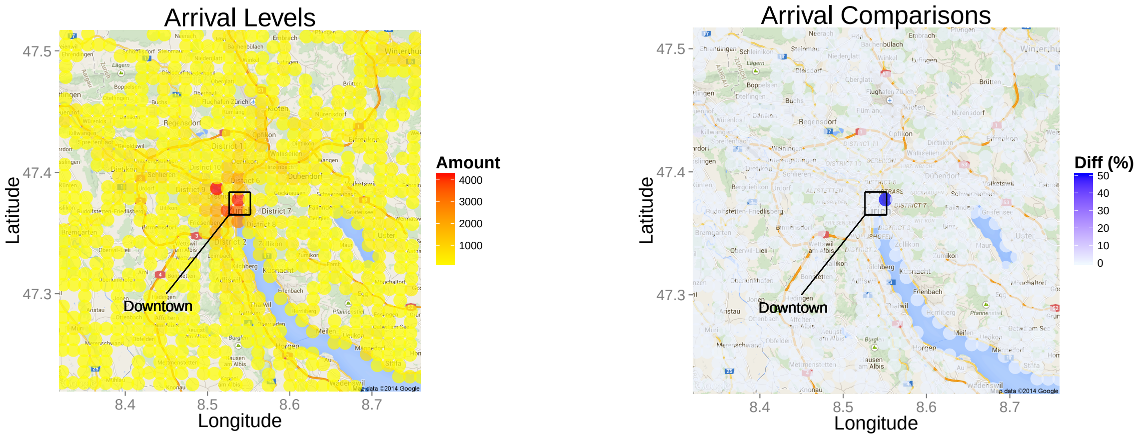
The same reasoning is valid regarding the arrivals. Figure 4.12 shows the validation results for downtown (Figure 4.12(a)) and for the suburbs (Figure 4.12(b)). Again, the CDFs generated agree with the existing data for both downtown and subur-



(a) Arrival CDF Comparison – Downtown

(b) Arrival CDF Comparison – Suburb

Figure 4.12. Arrival Validations



(a) Arrival Levels – Zurich

(b) Arrival Matrix Comparisons

Figure 4.13. Arrival Validations

ban regions. Furthermore, both 95% confidence intervals also include zero within their boundaries. Figures 4.13(a) and 4.13(b) present the expected arrival levels and the comparison of the generated and original O-D across the map, respectively. Numerically, practically 100% of the generated data differ only around 1% of the actual data. Thus, it is reasonable to state that the proposed model can also be used to generate arrival distribution for different scenarios.

To conclude, we validate the departures along each 10-minute interval, as depicted in Figure 4.14. Again, based on the CDF over-plot and on the confidence interval measures, we can argue that our proposed model was able to represent vehicle departures over time in Zurich. Since we focus here on the macroscopic aspects, the distance traveled by vehicles is modeled to help with microscopic definitions (i.e., departure and arrival points.). Thus, we do not validate this metric in the current work since it relies on the simulation environment.

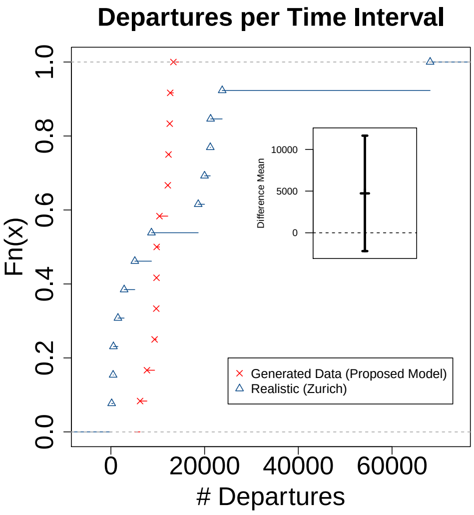


Figure 4.14. Departures Over Time Validation

4.4 Final Remarks

In this chapter, we present a characterization study about a large-scale vehicular mobility trace in terms of complex networks and O-D points. In summary, the results reveal that some vehicles are more likely to be better replicas than others. Moreover, it was possible to infer a model to represent macroscopic aspects related to O-D points. Given the difficulty to obtain a complete contact graph of a VANET and, consequently, the correspondent complex network metrics, we argue that O-D points, which are easily obtained from navigation systems, can be useful for helping the design of content delivery solutions. Therefore, in the next chapter we implement our VCDN model by considering the vehicles O-D points in the content replication module for two different scenarios: city-wide and region-wide.

Chapter 5

Content Delivery Solutions

As described in Chapter 1, the objective of this thesis is to investigate, to discuss, and to validate the hypothesis that the adoption of concepts from both CDN and P2P is effective to VANET applications with content delivery demand. In Chapter 2, we investigate how existing solutions have been explored the content delivery problem in VANETs. In Chapter 3, we propose the concept of Vehicular Content Delivery Network (VCDN) and a framework to implement this model. Finally, in Chapter 4, we characterize a large-scale mobility trace to obtain insights on how vehicular mobility may help on content delivery decisions.

In this chapter, we implement the VCDN model by following the proposed framework for two different applications with different demands. To this end, we take advantage of the characterization results by considering the origin-destination (O-D) points as basic input for our solutions. Given that replicating content is a fundamental issue to cost-effective content delivery in VANETs, we focus on the *Content Replication* module of the proposed framework. In fact, placing content where potential clients are expected to be will indeed reduce communication cost and delay, and consequently, improve the users' satisfaction.

In spite of validating our proposed VCDN model, the contribution of this chapter also lies in the advance in the state-of-the-art in concerns with content replication and delivery solutions for VANET applications. Differently from existing solutions, ours rely on vehicles' O-D points that are easily obtained from navigation systems installed on vehicles. In addition, we assess their performance under large-scale scenarios.

5.1 Methodology

In this section we describe some concepts and the basic characteristics of both city- and region-wide scenarios and evaluation.

5.1.1 Concepts

Connectivity Let V and S be the set of vehicles and infrastructure stations, respectively. Graph $G(V \cup S, E)$ represents the connectivity between vehicles, as well as between vehicles and infrastructure stations. An edge $e_{i,j} \in E$ indicates a contact between vehicle $v_i \in V$ and either another vehicle or an infrastructure station, $vs_j \in V \cup S$, where $v_i \neq vs_j$. Two arbitrary vehicles, or a vehicle and an infrastructure station, may have been through many different contacts along their lifetime. Therefore, the total duration $d_{i,j}$ of a contact $e_{i,j}$ is defined as the sum of duration of all those contacts.

Content A content C is a digital file of size C_s data units[1] that may be divided into C_c chunks of size $C_{cs} = \lceil \frac{C_s}{C_c} \rceil$ data units each. A content is considered valid for a period starting at t_c^i and finishing at time t_c^f. We refer to this period as the content's lifetime, within it must be delivered to the target vehicles.

Replication The content replication process aims at selecting a subset $R \subseteq V \cup S$ of vehicles or infrastructure stations to act as content replicas. Many strategies may be adopted to select the most appropriate vehicles, which depend on the application requirements and information about the vehicles. In this work, for example, we propose replica allocation solutions that rely on the origin-destination points of the vehicles.

Coverage A vehicle v_i may be covered basically by two different approaches: vehicle-to-vehicle (V2V) communication only, or with the help of vehicle-to-infrastructure (V2I) communication. Notice that V2V communication is preferred over V2I, since it is expected to cost less, as well as it should reduce the infrastructure workload. V2V and V2I present different communication capabilities, especially transmission rates, which affect the content delivery process. We define T_{V2V} and T_{V2I} the transmission rates, in data units per time unit, of V2V and V2I communications, respectively. Thus, the time required to transmit a content of size C_s using only V2V communication is

---

[1] We are using the general term *data units* to represent content in terms of its storage unit (e.g., bits, Bytes, and KBytes)

$\frac{C_s}{T_{V2V}}$. Then, a vehicle v_i is covered by V2V communication only if its total contact time with replica nodes is sufficient to receive the entire content:

$$\sum_{j \in R \cap V} d_{i,j} \geq \frac{C_s}{T_{V2V}} \tag{5.1}$$

Nevertheless, a vehicle may also be covered with the help of infrastructure stations. In this case, a vehicle v_i is considered to be covered if its total infrastructure contact duration is sufficient to transmit the remaining parts of the content that could not be transmitted by V2V contacts. This is defined as:

$$\sum_{j \in R \cap S} d_{i,j} \geq \frac{C_s - \frac{T_{V2V}}{\sum_{j \in R \cap V} d_{i,j}}}{T_{V2I}}. \tag{5.2}$$

Time to be Covered The time a vehicle has to wait before being covered is also an important metric. Thus, we define t_i^d as the time a vehicle v_i departs from its original point, and t_i^c as the time when it is covered. For many applications, the lower the waiting time, $t_i^c - t_i^d$, the better.

Delivery Cost To represent the cost of delivering a content C to a vehicle v_i, let C_{V2I} and C_{V2V} be the cost to transmit a data unit using infrastructure (V2I) and ad hoc (V2V) communication, respectively. In general, $C_{V2I} \gg C_{V2V}$. Then, the cost to transmit a content C to a vehicle v_i is represented by

$$C_c^i = C_s \times C_{V2V}, \tag{5.3}$$

when only V2V communication is sufficient, and by

$$C_c^i = \frac{T_{V2V}}{\sum_{j \in R \cap V} d_{i,j}} \times C_{V2V} + (C_s - \frac{T_{V2V}}{\sum_{j \in R \cap V} d_{i,j}}) \times C_{V2I} \tag{5.4}$$

when the infrastructure stations are part of the communication solution. Thus, the lower the number of transmitted messages through V2I communication, the better.

5.1.2 System Model

Figure 5.1 illustrates the assumed network architecture. We assume a hybrid scenario where vehicles are equipped with cellular as well as WAVE (Wireless Access for Vehicular Environment) [Morgan, 2010] communication modules. This is a reasonable assumption, since most vehicles are expected to be connected to the Internet in the

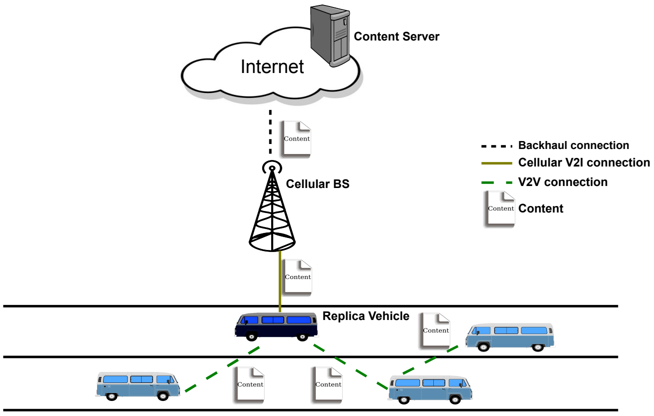


Figure 5.1. System model architecture. Vehicles are capable of communicating through cellular (V2I) and V2V networks. Replica vehicles deliver content using V2V only.

near future [Lu et al., 2014]. This is a trend supported by both academia [Barros, 2014] and industry [Telefonica, 2014]. Therefore, the system model we assume will make the deployment of our solution quite simple.

We adopt a centralized approach consisting of a content server with high computational capabilities in terms of memory, disk, and processing units. Because of commercial and technology constraints, the cellular communication (V2I) costs significantly more, in terms of money, than the V2V solution. This leads to the objective of giving priority to V2V over V2I communication, whenever possible.

Content is assumed to be static, and should be delivered to all vehicles in the network (city-wide scenario) or in a region of interest (region-wide scenario) during its lifetime. At departing time, vehicles use their navigation system to collect their O-D points, which are then sent to the server through the cellular network, encapsulated in an *ENTER* message. Based on the received information for each round, the server selects the vehicles expected to be good replicas. The server then sends, to the selected vehicles, the content encapsulated into a *CONTENT* message, also through cellular communication. Upon receiving the content, the replica vehicles periodically disseminate it to their peers, through V2V communication. It should be noted that, in the region-wide scenario, replica vehicles only disseminate content when inside the region of interest (RoI).

It should be clear that the selection of replicas may impose a communication cost, since replica vehicles will communicate through V2V link with their peers. This may lead to network congestion, bandwidth waste, packet loss, and, consequently, a poor quality of service. Furthermore, covering vehicles in a large-scale scenario is not a trivial task, given the mobility dynamics, and the diverse conditions faced by vehicles. In other words, there is a trade-off between coverage and cost. Therefore, one requirement of our proposed solutions is to be flexible to be used by a variety of applications with different demands in terms of content availability and communication cost.

To this end, we argue and validate that content can be effectively replicated using easily obtained information (i.e., O-D points) and low-cost algorithms, in contrast to the majority of solutions found in the literature. The assumption that vehicles know their destination points is reasonable given the increasing market penetration of online navigation systems, such as *Waze* [waz, 2015], in which users provide their destination even for known routes to avoid traffic congestion. The adoption of such systems is expected to increase even more with the advance of solutions that provide real-time traffic status [Younes et al., 2014]. Furthermore, context-aware navigation systems [Ramazani and Vahdat-Nejad, 2014] are expected to learn users' routine and infer their destination. In fact, *Waze* already does that and suggests a destination to the users depending on their current location and period of day.

5.1.3 Performance Evaluation Basic Configuration

We conduct extensive simulations to assess the performance of our solutions when compared with other existing ones. Given the large-scale nature of VANETs, it is important to evaluate our solutions under a large-scale mobility scenario. Due to the computational complexity of existing network simulators [Joerer et al., 2012], we perform evaluations through two complementary approaches: large-scale and network-enabled.

In the large-scale approach, we adopt a realistic large-scale mobility scenario to measure how our solutions perform when a large number of vehicles is involved, in an ideal network scenario, where packets are not lost and the communication conditions are always satisfied. To this end, we implemented a large-scale simulator in the R environment. The adopted trace [Uppoor and Fiore, 2011] encompasses over 120,000 vehicles traveling from 6:00 am to 8:00 am (i.e., 7200 s) of a week day in the city of Cologne, Germany, in a 400 km^2 area.

On the other hand, in the network-enabled approach, we adopt the OMNET++[2] network simulator, in which vehicles implement the WAVE (wireless access for vehicular environment) suite [Morgan, 2010] that includes the IEEE 802.11p standard [Jiang and

---

[2] http://www.omnetpp.org

Delgrossi, 2008] as MAC and physical layers and the IEEE 1609 protocol suite to define the upper-layer operations. In the network-enabled study, we adopt a realistic mobility scenario of the city of Ottawa, Canada, for the city-wide solution and of Manhattan, NYC, for the region-wide case. To improve the realism of the physical layer even more, we also adopt the shadowing model described in [Sommer et al., 2010]. This is a realistic model for urban environments based on IEEE 802.11p measurements, and simulates signal attenuation caused by buildings. To simulate realistic vehicle movements, we take advantage of the mobility model defined by SUMO[3].

5.1.3.1 Baseline Solutions

We compare each of our solutions with two others from the literature. For the city-wide case, we adopt as baseline solutions the Push-and-Track [Whitbeck et al., 2012] and Selective Flooding [Vahdat et al., 2000]. For the region-wide one, we adopt Push-and-Track as well and Linger [Fiore et al., 2013]. We describe each of them as follows.

Push-and-Track is a content delivery solution in which an infrastructure server, accessible through the cellular network, keeps track of all target vehicles already covered (i.e., received the content). A vehicle sends to the server an $ENTER$ message when it becomes a target for a content, and a $LEAVE$ message when it is no longer a target. By the time a vehicle receives the content, it sends an ACK message to the server. These messages are transmitted only through the cellular network. Based on the received ACK messages, the server is aware of how many vehicles are still uncovered. Furthermore, the server expects a linear coverage behavior, where at least $p\%$ of the target vehicles are expected to be covered after $p\%$ of the content lifetime has elapsed. If this coverage expectation is not satisfied, the server randomly selects new replicas and sends the content to them using the cellular network. The replicas deliver content periodically every δ seconds. Push-and-Track also defines a panic zone starting some time before the content expiration time, when all uncovered vehicles receive the content through the cellular network.

Selective flooding is well adopted in the literature and refers to the idea of vehicles selectively forwarding a content to their neighbors. By selectively, we mean that only vehicles expected to provide good dissemination performance and coverage will forward a content. Then, upon receiving a content, a vehicle decides whether to forward it to its neighbors. Different strategies may be used to decide forwarding a content, such as the ones based on the vehicle position, network density and randomness. In this work, we adopt a random approach in which vehicles only forward a

---

[3] http://sumo-sim.org

content with f_p probability. Furthermore, a parameter θ is used to indicate the number of vehicles to start the application acting as local providers. A server in this case is responsible for sending, through cellular network, the content to these vehicles.

Linger is a totally distributed solution in which vehicles compute a comparable index that indicates how suitable they seem to be as replicas. The index takes into account the vehicle's speed, direction, and distance to the RoI. The higher the index, the better replica a vehicle is expected to be. Initially, the vehicle that first senses or generates the geo-localized data to be shared is assumed as replica. From this time on, a distributed replica selection process starts with the first replica computing and sending its index to its one-hop distance neighbors. Upon receiving the index, these neighbors also compute their own indices, and compare them with the received value. When the computed index is higher than the received one, the replica candidate waits for a period that is inversely proportional to its index before responding to the current replica, indicating that it should become a replica. Thus, the first vehicle that responds is expected to have the higher index among the neighbors, and is then selected as the new replica. It should be noted that many replicas may exist simultaneously, and that each is responsible for looking for new, better replicas. Each replica delivers content to its neighbors every δ seconds. Here, we consider δ to be 1 s, which is the same value used for the other solutions.

5.1.3.2 Evaluation Metrics

We measure two major metrics to evaluate the performance of our solutions: content availability and delivery cost. The former refers to the ease of content availability to interested vehicles, in terms of coverage, time to be covered, and capacity. The coverage is defined as N_c/N_t, where N_t is the number of target vehicles and N_c is the number of target vehicles covered. The time to be covered for vehicle v_i is computed as $t_i^c - t_i^d$, where t_i^d is the time when v_i departs from its origin point, and t_i^c is the time when it is covered. Finally, the capacity represents the amount of data that could be transmitted by replica vehicles, and is computed as $\sum_{i,j} d_{i,j} \times T_{V2V}\ \forall i \in R,\ j \in V$, where $d_{i,j}$ is the duration of contacts between v_i and v_j, R is the set of replica vehicles, V is the set of all vehicles, and T_{V2V} is the transmission rate of the V2V communication technology.

The latter major metric, delivery cost, refers to the number of messages exchanged, and the amount of redundant messages that is transmitted. The number of V2I or infrastructure messages N_{V2I} is the number of all messages exchanged between infrastructure stations and vehicles (i.e., Enter, Leave, Ack, content from the server to selected replicas). The number of ad hoc or V2V messages N_{V2V} is the total number of messages exchanged between vehicles (i.e., control messages in Linger, content from replicas to vehicles). The redundant messages for a vehicle v_i is defined as $R_i = NC_i - 1$, where NC_i is the number of times v_i received the same content. The total of the redundant messages is then the sum for all vehicles, defined as $R_{total} = \sum_{i \in V} R_i$. In addition, we measure the number of network lost packets for the network-enabled study to assess how each solution affects network performance.

In summary, the higher the coverage and the capacity, and the lower the time to be covered, the redundancy, and the cost in general, the better.

5.2 City-wide Scenario

The main objective of this solution is to propose and evaluate a city-wide content delivery solution for VANETs in which all vehicles in the network must be covered with the content. Based on the vehicles' origin-destination (O-D) points, the proposed solution, called **O**rigin-**D**estination-based **C**ontent **R**eplication (ODCRep), is flexible to be applied to different applications, and focuses on balancing the number of replicas in the entire city. To evaluate our proposal, we compare it with two existing solutions named Push-and-Track [Whitbeck et al., 2012] and Selective flooding [Vahdat et al., 2000], by running exhaustive simulations and measuring content availability and delivery cost.

5.2.1 ODCRep: Origin-Destination-based Content Replication

The implementation of ODCRep follows the VCDN model and framework proposed in this thesis, with focus on the content replication module. Given the focus of this thesis, we do not consider any implementation for the incentive control module, which is out of scope of our work.

5.2.1.1 Content Replication and Management

The ODCRep content replication process runs in the server, which is responsible for deciding whether an entering vehicle should become a replica. Given that it is not possible to know, in advance, which vehicles will travel for longer periods, ODCRep estimates the probability of a vehicle being a good replica based on current information only. Three metrics contribute to this probability: the vehicle's estimated travel time, the content lifetime, and the vehicle's departure and arrival locations. Notice that these metrics are easily obtained from the content specification, and from the O-D points. Furthermore, each contribution can be parametrized according to the application's demand in terms of cost and content availability. In the following, we describe how each metric contributes to the selection of replica vehicles.

Travel time contribution The vehicles' travel time is an important metric, as presented in Chapter 4. ODCRep aims at selecting the vehicles expected to travel for longer periods. As result, we expect high content availability, since those vehicles can contribute significantly in the delivery process.

To achieve this goal, the ODCRep server receives and stores all vehicles' estimated travel time. A vehicle is then considered a replica candidate only if its travel time is higher than a threshold, which separates all estimations into two groups: the first one where $\phi\%$ of the values are smaller, and the second one where $(100 - \phi)\%$ are higher than the threshold, considering $\phi \in [0, 100)$ is an application-defined value. Vehicles expected to travel for periods shorter than the threshold are not even considered as replica candidates.

An important issue here is how to efficiently compute the threshold considering that the application is expected to operate in large-scale scenarios. To this end, the estimated travel time of all vehicles are stored in two heap data structures called *maxHeap* and *minHeap*. The *maxHeap* stores the $\phi\%$ of the smallest values, while the *minHeap* stores the $(100 - \phi)\%$ highest values. Both heaps satisfy the heap property meaning that the first element of *minHeap* is the smallest value among the $(100 - \phi)\%$ highest ones. Similarly, the first element of *maxHeap* is the highest value among the $\phi\%$ smallest ones. Then, the threshold is easily picked as one of the first element of any heap.

In terms of computational complexity, picking the threshold takes $\mathrm{O}\big(1\big)$ operations, given that the heaps satisfy the heap property. To satisfy the heap property, an insertion takes $\mathrm{O}\big(\log\ n\big)$ operations. In addition, after a number of new inserted values, the heaps must be balanced to keep $\phi\%$ of the values in the *maxHeap*, and $(100 - \phi)\%$ of them in the *minHeap*, which also takes $\mathrm{O}\big(log\ n\big)$. Thus, the overall process requires $\mathrm{O}\big(\log\ n\big)$ operations.

The parameter $\phi\%$ is used to balance content availability and cost. The highest this value is, the less likely for an arbitrary vehicle to be considered as replica candidate. Consequently, a small number of replicas may be selected, which may reduce the content availability and save communication resources. In contrast, more vehicles may be selected as replicas when $\phi\%$ is low. Thus, the application designer should set $\phi\%$ based on the application demands.

Notice that the threshold approximates the real value (i.e., the threshold for all vehicles) as new vehicles enter the network. Therefore, this value may be not accurate in the beginning of the application, which may lead to the selection of not so good replicas. To help ease this problem, other metrics also contribute to the replica selection, as described next.

Content lifetime contribution The purpose of ODCRep is to select replica vehicles to help deliver content to all vehicles traveling during the content's lifetime. Therefore, the content expiration time plays an important role on the replica selection process.

City-wide content delivery applications may demand different delivery coverage, relative to the content lifetime. While some applications may require that all vehicles, no matter their departure time, must be covered, others may be more flexible, and are not affected by uncovered vehicles departing close to the content's lifetime expiration. Therefore, our solution should be compliant with this diversity of application demands.

To this end, the content lifetime contribution has three categories. The first one represents the period encompassing less than 25% of the content lifetime. The second category represents the period between 25% and 75% of elapsed lifetime. Finally, the last one represents the period from 75% of elapsed time to the expiration time. Thus, the application's designer can configure the content lifetime contribution according to its expected delivery over time, as explained below.

Let $t_e = \frac{t - t_c^i}{t_c^f - t_c^i}$ be the fraction of time elapsed after the content lifetime began, where t is the current time, and t_c^i and t_c^f define the content lifetime interval. Thus, we define a linear function $w_t(t_e)$ representing how this metric contributes to the selection process in terms of the percentage of the elapsed time, t_e, after the content lifetime begins:

$$w_t(t_e) = \begin{cases} \alpha_t \times t_e + \beta_{t1} & \text{if } 0 < t_e < 0.25 \\ \alpha_t \times t_e + \beta_{t2} & \text{if } 0.25 \leq t_e < 0.75 \\ \alpha_t \times t_e + \beta_{t3} & \text{if } t_e \geq 0.75 \end{cases}$$

where α_t indicates the changing rate in the function according to t_e, and $\beta_{t1} < \beta_{t2} < \beta_{t3}$ are the coefficients.

The parameters of this function should also be used to balance content availability and communication cost, based on the application's demand. The function $w_t(t_e)$ is flexible enough to achieve different application demands in terms of content availability over time. The parameter α_t defines the weight of this metric to the replica selection. In addition, the parameters β_{t1}, β_{t2}, and β_{t3} define the importance of each period of the content lifetime in the replica selection. For example, when content should be delivered to all vehicles, no matter their departure time, all of the three parameters should have an identical, high value. On the other hand, when content is not useful at the ending of its lifetime, the β_{t3} parameter should be assigned to a negative high value, so less vehicles will be selected as replicas in this period.

Departure and arrival areas contribution The vehicle's departure (origin) and arrival (destination) areas also play a determining role in the replica selection. Since ODCRep focuses on city-wide scenarios, it attempts to balance the replica placements across the entire map to increase the content availability. Otherwise, unbalanced allocated replicas may lead to uncovered and over-covered areas.

Thus, we take advantage of the results described in Chapter 4, where we determine how the number of departures and arrivals are distributed across downtown and suburb regions, in terms of their demographic density. Based on those results, we estimate the number of vehicles departing from and arriving at each quadratic area of 1000-meter side, and then we calculate the number of vehicles to be selected as replica for each area as a percentage θ of the total estimated.

To this end, let d_k and a_k be the estimated number of vehicles departing from and arriving at area k, respectively. For each of the vehicles departing from or arriving at area k, we want to select up to $\theta\%$ of them as content replicas, according to the application demands. In addition, let d_k^r and a_k^r be the number of already selected replicas departing from and arriving at area k, respectively. Thus, $d_k^r \leq d_k \times \theta$ and $a_k^r \leq a_k \times \theta$.

We define two functions, $w_d(k)$ and $w_a(j)$, to calculate the weight to be assigned to a vehicle in terms of its departure area k, and its arrival area j, as:

$$w_d(k) = -\alpha_d \times (\frac{d_k^r}{d_k \times \theta} \times 100) + \beta_d \tag{5.5}$$

and

$$w_a(j) = -\alpha_a \times (\frac{a_j^r}{a_j \times \theta} \times 100) + \beta_a \tag{5.6}$$

where constants α_d and α_a must be in $(0, 1]$ range, and β_d and β_a must be assigned a value in the $(0, 100]$ range.

In summary, since both functions $w_d(k)$ and $w_a(j)$ are negatives, the higher the number of already selected vehicles departing from or arriving at an area, the lower the chance of a vehicle to be selected. The parameter θ is also used to help balance the number of replicas over space and time. A higher value will lead to more replicas, and, consequently, higher communication costs. A lower value, on the other hand, will reduce communication costs, as well as content availability. Therefore, θ should be assigned to a value to meet the application demands.

Algorithm 1 - Server Replica Selection

Input: ENTER message msg from vehicle v (containing its travel time $travelTime$, departure area $dArea$ and arrival area $aArea$), $maxHeap$ and elapsed content lifetime t_e

Output: CONTENT message only if v is selected as replica.

```
procedure IsReplica(v)
    $threshold \leftarrow maxHeap[1]$
    if $msg.travelTime \geq threshold$ then                ▷ Is $v$ a replica candidate?
        $wLifetime \leftarrow w_t(t_e)$
        $wDeparture \leftarrow w_d(msg.dArea)$
        $wArrival \leftarrow w_a(msg.aArea)$
        $probReplica \leftarrow (wLifetime + wDeparture + wArrival)/3$
        if $probReplica \geq Random(0, 100)$ then    ▷ $v$ is replica with probability $probReplica$
            send CONTENT message to $v$
        end if
    end if
end procedure
```

Putting all together After defining the contributions of the vehicles' travel time, content lifetime, and vehicles' origin-destination areas to the replica allocation process, we now put them together. Algorithm 1 shows the procedure running in the server after receiving an *ENTER* message from vehicle v.

5.2.1.2 Content Discovery and Delivery

The vehicles selected as replicas must act as local providers, helping delivering content to their peers using V2V communication only. Given this is a push-based application, the content discovery module is not required. To this end, upon receiving the content from the server, a replica vehicle delivers it periodically to its peers, every δ seconds. Given the dynamic topology of VANETs, a contact between a replica vehicle and a client interested in content may last for a short period. Therefore, the value for parameter δ should be chosen to increase the chance of a client to be covered, even for a short duration contact with the replica. On the other hand, notice that the lower the δ, the higher the number of messages.

5.2.1.3 Mobility Service

The mobility information used in this solution is the O-D points of vehicles. Therefore, this module is responsible for collecting the departure and arrival points and make them available to the vehicle that, as described before, send them to the server through the cellular network.

5.2.2 Performance Evaluation

5.2.2.1 Simulation Setup

ODCRep was proposed to operate in a hybrid architecture model where vehicles are able to communicate through cellular network (V2I), as well as among themselves using ad hoc communication (V2V). For the latter, the IEEE 802.11p standard is assumed to be implemented in all vehicles. The entire map is assumed to be covered by the cellular network that is able to transmit data at a rate up to $T_{V2I} = 1\,\text{Mbps}$. We assume V2V and V2I transmission rates to be of 1 Mbps and 3 Mbps, respectively, based on real experiments conducted in [Teixeira et al., 2014]. Vehicles have a transmission range of 100 m based on the results presented in [Cheng et al., 2007]. When it comes to transmission cost, it is assumed that $C_{V2I} \gg C_{V2V}$. Therefore, the objective is, in general, to reduce the V2I communication, which is the most expensive one.

The application aims at delivering a content of size $C_s = 100\,\text{KBytes}$ represented by a single fragment of the same size. Thus, a contact lasting for one second is sufficient to the entire delivery. To evaluate the performance in long running applications, the content lifetime is assumed to last for the entire running period of the application. Given the city-wide coverage demand, all vehicles are expected to be covered.

Table 5.1 presents the ODCRep specific parameters, as well as the ones for the baseline solutions, configured for the simulation studies. These values were chosen with the objective of balancing the trade-off between content availability and communication cost. To demonstrate the flexibility of our solution, we suppose a low effect on the application's performance for uncovered vehicles departing close to the content expiration lifetime. This is represented by the negative, high value for parameter β_{t3}. The other values were chosen after plotting and observing the behavior of the functions $w_t(t_e)$, $w_d(k)$, and $w_a(j)$ under different inputs.

5.2.2.2 Large-scale study

The objective here is to measure how ODCRep performs under a large-scale scenario. We vary θ from $[0.1, 0.5]$ (i.e., $[10\%, 50\%]$) to evaluate the impact of the number of

Table 5.1. Simulation Configuration Parameters.

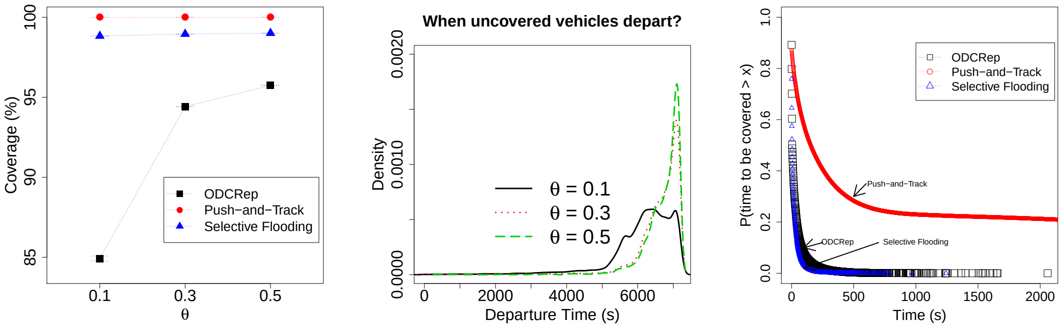
Parameter	**Value**
ϕ	98.0%
α_t	4.0
β_{t1}	0.0
β_{t2}	100.0
β_{t3}	-400.0
α_a	0.8
β_a	80.0
α_d	0.8
β_d	80.0
δ	1 second
f_p (Selective Flooding only)	0.1
Panic Zone (Push-and-Track only)	10 s prior to the end
θ	[10%, 50%] for large-scale study 30% for network-enabled study
Simulation time	7200 s for large-scale study 3600 s for network-enabled study
Simulation area	400 km^2 for large-scale study 9 km^2 for network-enabled study

replicas from each departing and arriving areas. All results represent the average and the 95% confidence interval from 33 simulations.

Figure 5.2 presents the content availability results. Notice in Figure 5.2(a) that, the higher the θ, the higher the coverage for ODCRep, since more vehicles are selected as replicas, according to functions $w_d(k)$ and $w_a(j)$. In addition, the coverage achieved by ODCRep reaches 96% when $\theta = 50\%$. Not surprisingly, Push-and-Track achieved 100% coverage, since all uncovered vehicles are covered by the server during the panic zone. Selective Flooding also achieved high coverage, because of its flooding spread mechanism. However, there is a cost to be paid by those solutions, as discussed later.

Figure 5.2(b) demonstrates the flexibility of our solution. As stated earlier, we configured ODCRep to be flexible in the delivery for late departing vehicles, by setting the parameter β_{t3} to a high negative value. In fact, this demand was accomplished, as shown in Figure 5.2(b), since most uncovered vehicles depart after 75% of the content lifetime has elapsed.

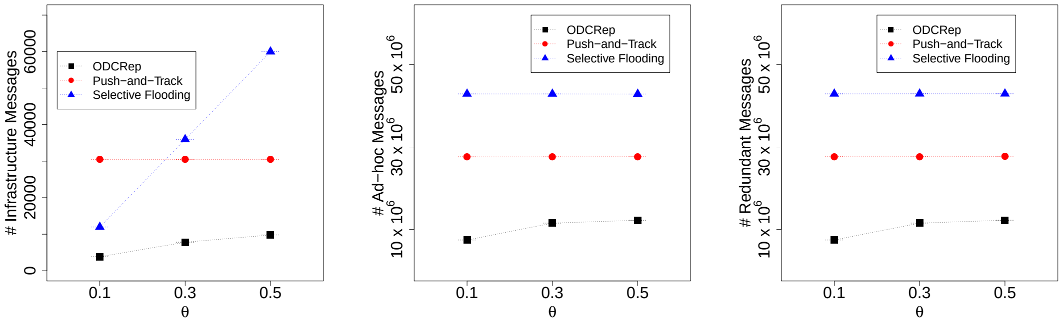
Another important metric is the time to be covered, which measures how long a vehicle waits before being covered. Figure 5.2(c) shows the Complimentary Cumulative Distribution Function (CCDF) of the waiting time of all covered vehicles, assuming $\theta =$



(a) Coverage (b) ODCRep Uncovered Vehicles Departure Time (c) Time to be Covered

Figure 5.2. Content availability results for the network-enabled study. ODCRep achieved good coverage results together with low time to be covered. In addition, ODCRep's flexibility is demonstrated since most uncovered vehicles depart close to the end of the content lifetime.

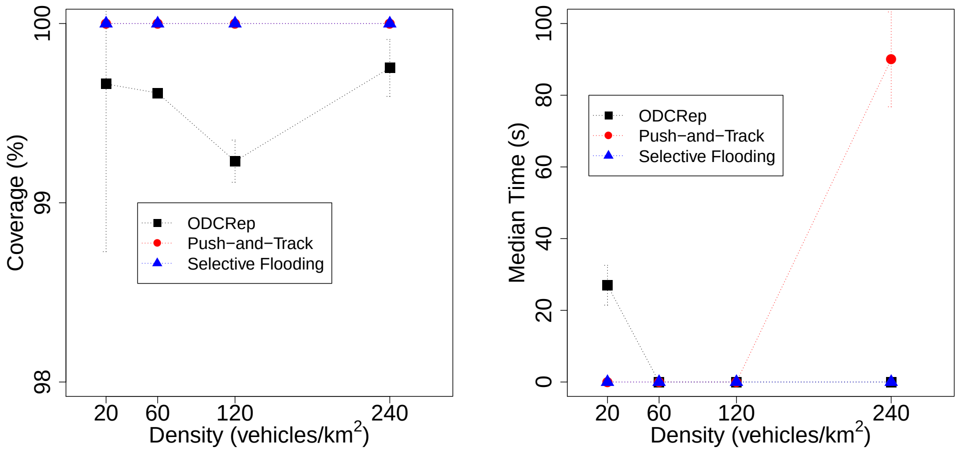
0.5. Notice that vehicles in ODCRep and Selective Flooding have higher probabilities of being quickly covered, when compared to Push-and-Track. In fact, many vehicles in Push-and-Track are only covered in the panic zone period, due to its unbalanced replica allocation. Again, the balanced replica allocation achieved by ODCRep leads to a higher chance of uncovered vehicles to be in contact with replicas. This is also true in Selective Flooding.



(a) Infrastructure (V2I) Messages (b) Ad hoc (V2) Messages (c) Redundant Messages

Figure 5.3. Delivery cost in terms of messages exchanged and redundant messages for the large-scale study. OCDRep outperformed the baseline solutions since it balances the number of replicas over time and space.

The communication cost is also an important metric, particularly when it comes to V2I (i.e., infrastructure) transmissions, which is more expensive than the V2V (i.e., ad hoc) one. Since *ENTER* (Push-and-Track and ODCRep), *LEAVE* and *ACK* (Push-and-Track) messages are relatively short, when compared to content, they are not



(a) Coverage for network-enabled study.

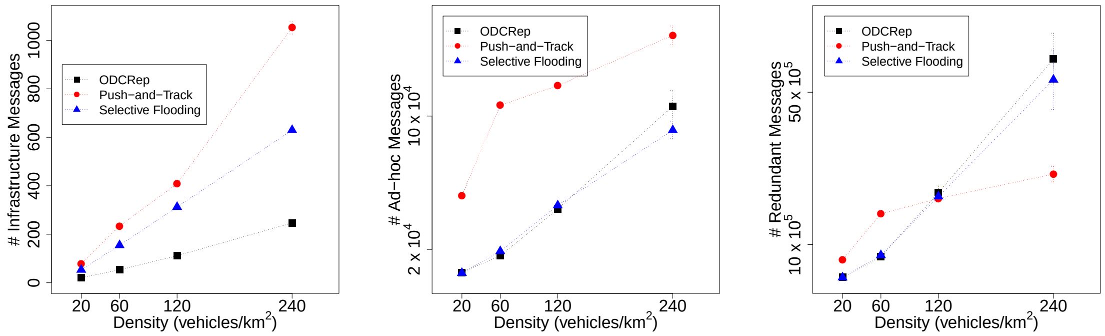
(b) Time to be covered for network-enabled study.

Figure 5.4. Coverage and time to be covered results for the network-enabled scenario.

considered here for the sake of simplicity. However, it is known that ODCRep and Push-and-Track require as many *ENTER* messages as the number of vehicles, and Push-and-Track requires as many *ACK* messages as the number of covered vehicles. As illustrated in Figures 5.3(a) and 5.3(b), ODCRep requires less V2I and V2V messages than the baseline solutions. In addition, θ does not affect significantly the results of ODCRep. Notice that the higher the θ, the higher the number of infrastructure messages exchanged in Selective Flooding, since the number of initial replica vehicles increases with θ.

Redundant messages refer to duplicated messages received by vehicles, which lead to a waste of network resources. The number of redundant messages is significantly higher for Selective Flooding and Push-and-Track, when compared to ODCRep, as illustrated in Figure 5.3(c). ODCRep reduces the number of redundant data by balancing the content replication according to vehicles' departing and arriving areas. On the other hand, Selective Flooding and Push-and-Track do not use such information, which may lead to unbalanced replica allocation.

In general, we can state that ODCRep could deliver content quickly to a high number of target vehicles (around 96%), by consuming significantly less network resources than Push-and-Track and Selective Flooding.



(a) Infrastructure (V2I) messages

(b) Ad hoc (V2V) messages

(c) Redundant messages

Figure 5.5. Delivering cost in terms of messages exchange for the network-enabled study.

5.2.2.3 Network-enabled study

The objective here is to evaluate ODCRep when vehicular specific network protocols are used. We run simulations under low, medium, and high network density to evaluate different scenarios. All results represent the average and the 95% confidence interval from 33 simulations.

The coverage, shown in Figure 5.4(a), was nearly 100% for all solutions. ODCRep achieved over 99% of coverage for all density scenarios (from 20 to 240 $vehicles/km^2$). Similar to the large-scale results, both Push-and-Track and Selective Flooding were able to achieve practically 100% of coverage, no matter the network density. In terms of time to be covered (Figure 5.4(b)), our solution was able to deliver content to target vehicles as soon as they enter the network for all scenarios, except for the lower-density one (20 $vehicles/km^2$). This is due to the fact that ODCRep balances the replicas over time and space, leading to uncovered areas in the ending of the application. In contrast, Push-and-Track leads to very high time to be covered for the high-density scenario (240 $vehicles/km^2$) because of its unbalanced replica allocation. In fact, many vehicles were only covered during the panic zone period. Selective Flooding, on the other hand, was very effective in this metric no matter the network density.

ODCRep also required less V2I messages than the baseline solutions, as shown in Figure 5.5(a). This is an important result, given that V2I transmissions are more expensive than V2V. In addition, the higher the number of vehicles, the higher the number of V2I messages, since more replicas are selected. This is also true for the Selective Flooding approach. Push-and-Track performed poorly in this metric, as well as the number of V2V exchanged messages, as shown in Figure 5.5(b). Again, this is due to the fact that replicas are selected in a way that they are balanced across the areas. On the other hand, Push-an-Track requires more messages because of its replica allocation based on the current coverage, and also due to unbalanced selection.

The amount of redundant messages increases with the number of vehicles for all solutions, as illustrated in Figure 5.5(c). Push-and-Track presents the worst results when it comes to redundant messages for lower-density scenarios, due to its unbalanced replica allocation. However, Push-and-Track reduces the number of redundant messages for the higher-density scenario because many vehicles are only covered during the panic zone period by V2I communication.

5.3 Region-wide Scenario

We also propose and evaluate a **G**eo-Localized **O**rigin-**D**estination-based **C**ontent **R**eplication (GO-DCR) solution designed for VANET applications. GO-DCR relies on the vehicles' origin-destination (O-D) points (i.e., departure and arrival points) to select those that are more likely to be effective in keeping content inside a RoI. To evaluate our proposal, we compare it with two existing solutions named Push-and-Track [Whitbeck et al., 2012] and Linger [Fiore et al., 2013], by running exhaustive simulations and measuring content availability and delivery cost.

5.3.1 GODCR: Geo-localized Origin-Destination Content Replication

The objective of this application is to keep content available inside a circular region of interest (RoI), so all vehicles travelling through it will be covered. Content is assumed to be static and is provided by a vehicle that has sensed geo-localized useful information, such as a traffic jam situation. This vehicle uses a cellular network to send the sensed content and its location reference to the server, which then proceeds to the execution of the replication task. At time of departure, ordinary vehicles send a message containing their O-D points, also using the cellular network.

The implementation of GODCR follows the VCDN model and framework proposed in this thesis, with focus on the content replication module. Given the focus of this thesis, we do not consider any implementation for the incentive control module, which is out of scope of our work.

5.3.1.1 Content Replication and Management

Different approaches can be used to strategically select replicas. In this work, we use the origin o_i and destination d_i points of vehicle v_i to decide whether or not it is expected to be a good replica. Furthermore, our solution measures a coverage index that indicates how well the RoI is covered by replicas over time, in order to decrease delivery cost.

The GO-DCR content replication process runs every σ seconds in the server. For each execution, the server evaluates the efficiency of vehicles that sent their O-D points after the last execution round for their roles as replicas. A vehicle is only considered a replica candidate if it is expected to pass through the RoI, based on its O-D points.

A vehicle v_i is considered to travel through the RoI when a straight line segment from its origin o_i to its destination d_i intercepts the circular area A that represents the RoI. We assume vehicles travel following a straight-line segment from their origin to their destination to save resources and time. In this case, there is no need to compute the vehicle's route and trajectory, which requires complex graph algorithms. Although some vehicles may be incorrectly selected as replicas even when they do not travel through the RoI, we argue that this is acceptable due to conservation of time and resources. To demonstrate this rationale, we compute the ratio between the number of vehicles that travel through the RoI considering their real trajectory over the number of those that are assumed to do so in a straight line. In other words, To this end, we consider 33 randomly generated RoI of radius 500 m, 1000 m and 2000 m for the Cologne scenario [Uppoor and Fiore, 2012, 2011] comprised of over 120,000 vehicles in a 400 km^2 area. In fact, as shown in the box plots of Figure 5.6, the precision is quite good, particularly for larger RoIs. Therefore, only a small number of replica candidates, based on the straight line, fail to travel through the RoI in their real trajectories.

Two metrics contribute to the replica selection in GO-DCR:

- the distance d_i^A a vehicle v_i travels inside the RoI A: the longer the better, because vehicles that travel for longer distances inside the RoI are more prone to cover more vehicles;
- a coverage index I_i^A for the interval starting at time t_i^e when v_i enters A and ending at t_i^l when it leaves. This is inversely proportional to the number of vehicles covering the RoI in the interval $t_i^e - t_i^l$: the higher the better, because fewer vehicles will be simultaneously covering A.

Figure 5.7 depicts examples of six possible scenarios of departure and arrival points. In the following, we refer to this figure whenever necessary in helping understand the solution.

To compute the distance that a vehicle travels inside A, we define the line formula that represents the segment $\overrightarrow{o_i d_i}$ as $y = mx + c$. Also, it is known that a circle A centered at point (p, q) with radius r is defined as $(x^2 - p) + (y^2 - q) = r^2$. Next, we find the interception points of the line in the circle, if any. To this end, we first compute the values of m and c by replacing the o_i and d_i coordinates into the line formula. Next, we replace y from the line formula into the circle formula, obtaining $(x^2 - p) + ((mx + c)^2 - q) = r^2$. To conclude, we then solve the quadratic equation to find the points (x_i^1, y_i^1) and (x_i^2, y_i^2) where the line intercepts the circle.

In the event that no results are found, the vehicle will not pass through A, and is then not considered as a replica candidate, as seen in the example of Vehicle 6 in Figure 5.7. Also, it should be noted that having interception points does not imply that the vehicle travels through A, since the line $y = mx + c$ is an extension of the segment $\overrightarrow{o_i d_i}$, as shown by Vehicle 5 in Figure 5.7. Although the line representing the segment $\overrightarrow{o_5 d_5}$ intercepts A, the segment itself does not. We then compare the segment $\overrightarrow{o_i d_i}$ to the points where the line intercepts the circle to check whether or not $\overrightarrow{o_i d_i}$ is inside A partially or totally. The distance d_i^A is then computed as the Euclidian distance of points (x_i^1, y_i^1) and (x_i^2, y_i^2) which is the part of $\overrightarrow{o_i d_i}$ that is indeed inside A. After this procedure, we have d_i^A for Vehicle v_i, as marked for Vehicles 1, 2, 3, and 4, in Figure 5.7. We then compute the contribution of this distance to the vehicle's

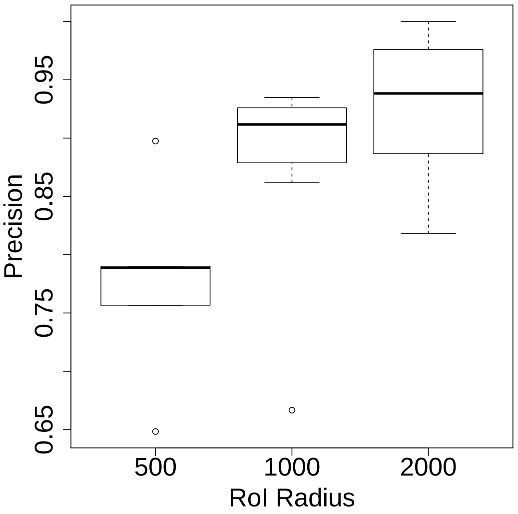


Figure 5.6. Precision of vehicles that indeed travel through a RoI when a straight line segment from its origin to its destination intercepts the RoI.

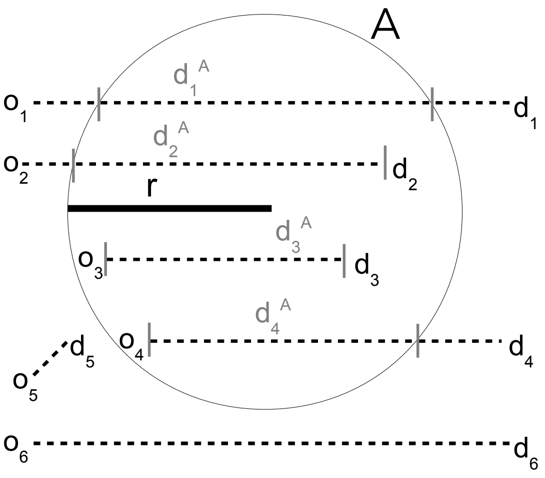


Figure 5.7. Different scenarios of vehicles intercepting the area A based on a straight line from their O-D points.

selection as the percentage of this distance relative to A's diameter:

$$w_i^d = \frac{d_i^A}{2 \times r}. \tag{5.7}$$

The next step is to compute the coverage index I_i^A that measures how well A is covered during the period Vehicle v_i is expected to be inside it. To this end, let A_v be the area covered by each vehicle, which is given by the V2V communication range. We then estimate the number n_i^A of replica vehicles that will be inside A simultaneously with v_i, and the total coverage area that they are able to achieve in the best scenario (i.e., with no overlaps). Since the best scenario has a very small chance of happening, we expect the occurrence of redundant coverage areas. Then, we set the index as a negative quadratic function:

$$I_i^A(n_i^A) = (\frac{-1}{10}) \times (\frac{n_i^A * A_v}{A})^2 + 100. \tag{5.8}$$

The reasoning is to have a higher chance of a vehicle being a replica when A is under cover. The idea of using a negative quadratic function is to be more flexible under a low number of simultaneous replicas, and more rigid in a selection already covered by numerous vehicles.

Finally, we compute the probability of a vehicle's selection as a replica by averaging both values defined by Equations (5.7) and (5.8):

$$p_i = \frac{w_i^d + I_i^A}{2}. \tag{5.9}$$

In summary, vehicles that travel for longer distances inside the RoI, when few other vehicles are expected to be there, are more prone to have higher selection probabilities.

5.3.1.2 Content Discovery and Delivery

The vehicles selected as replicas must act as local providers, helping delivering content to their peers using V2V communication only. Given this is a push-based application, the content discovery module is not required. To this end, upon receiving the content from the server, a replica vehicle delivers it periodically to its peers, every δ seconds, when inside the RoI. Given the dynamic topology of VANETs, a contact between a replica vehicle and a client interested in content may last for a short period. Therefore, the value for parameter δ should be chosen to increase the chance of a client to be covered, even for a short duration contact with the replica. On the other hand, notice that the lower the δ, the higher the number of messages.

5.3.1.3 Mobility Service

The mobility information used in this solution is the O-D points of vehicles. Therefore, this module is responsible for collecting the departure and arrival points and make them available to the vehicle that, as described before, send them to the server through the cellular network.

5.3.2 Performance Evaluation

5.3.2.1 Simulation Setup

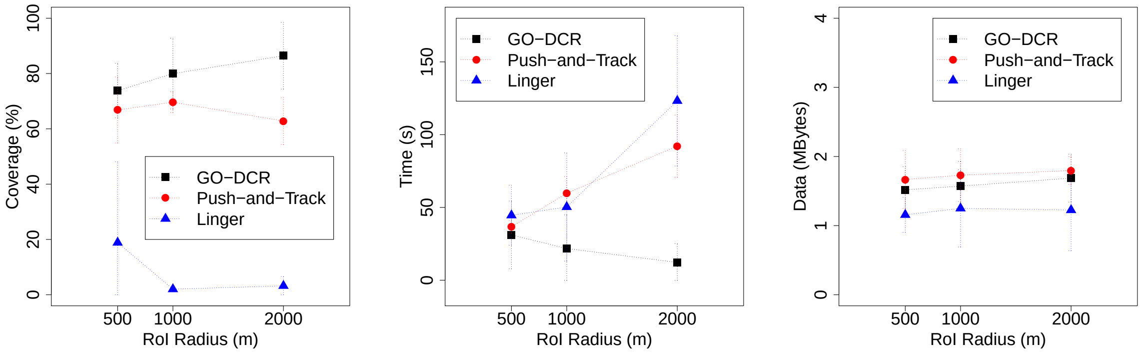
To assess the performance of GO-DCR as compared to two existing solutions, we conduct extensive simulations following two complementary approaches. The first, referred to as a large-scale study, adopts a large-scale mobility model and assumes an ideal network scenario with guaranteed packet delivery. In addition to the large-scale evaluation, we also implement and compare both solutions in the OMNET++ network simulator, which is referred to as a network-enabled study.

We assume V2V and V2I transmission rates to be of 1 Mbps and 3 Mbps, respectively, based on real experiments conducted in [Teixeira et al., 2014]. Vehicles have a transmission range of 100 m based on the results presented in [Cheng et al., 2007]. Hence, $A_v = \pi \times 100^2$ is the area covered by each vehicle. We consider a static content of size 100 Kbytes, as a single fragment of the same size, to be delivered to vehicles that are inside the RoI during the 2 hours of the mobility scenario. For all solutions, the replicas deliver the content periodically every δ =1 s when inside the RoI, and the server runs its allocation process every σ = 1 s. In Push-and-Track, the panic zone starts 10 seconds before the content lifetime expires. The number of simultaneous replicas assumed for Linger is 200 for the large-scale, and 4 for the network-enabled studies.

We present the simulation details and results for each study as follows. All results represent the mean and the 95% confidence interval of 33 simulation runs. For each run, a random RoI position is used, which is the same for all three solutions. This way, the solutions are compared under the same conditions, and for different scenarios. It is important to state that a different seed is used for the random number generator for each simulation run.

5.3.2.2 Large-scale study

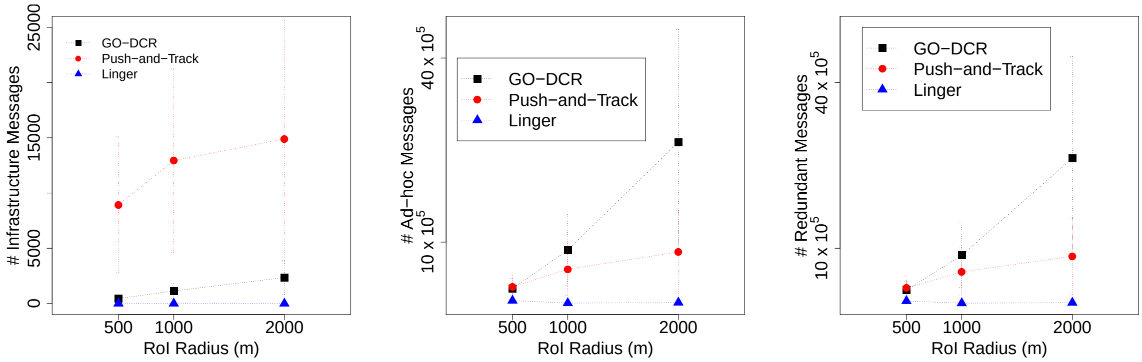
We vary the RoI radius from 500 m to 2000 m to measure how well the solutions perform under different application demands.



(a) Coverage (b) Time to be Covered (c) Capacity

Figure 5.8. Large-scale simulation results in terms of content availability. GO-DCR achieves higher coverage and lower time to be covered results, when compared to Push-and-Track and Linger.

Figure 5.8 presents the content availability results. GO-DCR covered more vehicles than Push-and-Track and Linger, as shown in Figure 5.8(a). This is because GO-DCR balances the number of replicas over time, and selects vehicles that are expected to be more valuable in terms of coverage. The coverage of Push-and-Track decreases for the 2000 m RoI radius scenario. In contrast, GO-DCR adapts accordingly to larger RoIs, because of the computed coverage index. It should be noted that Linger presents very poor coverage results in large-scale evaluations, particularly for larger RoI radius. Our network-enabled study confirms the proof provided by Linger's



(a) Cost – V2I Messages (b) Cost – V2V Messages (c) Redundancy

Figure 5.9. Large-Scale simulation results in terms of communication cost. GO-DCR leads to a low number of V2I messages. However, the higher coverage achieved led to a higher redundant, in terms of V2V messages.

authors of its effectiveness for small mobility scenarios. However, Linger performs more poorly when large-scale mobility scenarios are used. In fact, given the large number of vehicles to be covered, Linger cannot select the appropriate replicas, covering only a small subset of all target vehicles. It is important to state that not even the panic zone strategy of Push-and-Track could improve its coverage; many vehicles leave the RoI without being covered, and then are not considered in this panic stage. Thus, this strategy is only effective for vehicles that are inside the RoI when the panic zone starts.

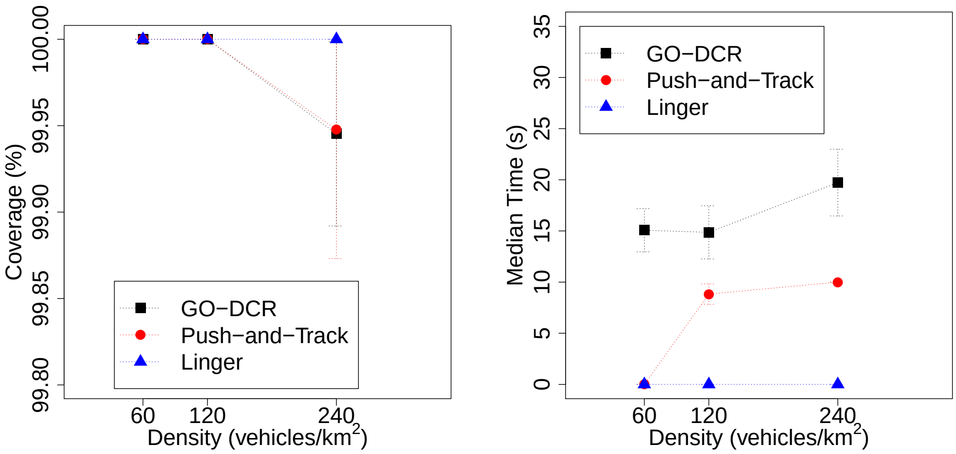
The appropriate replica allocation also led GO-DCR to cover vehicles earlier than Push-and-Track and Linger, as shown in Figure 5.8(b). This metric is of great value when content must be delivered as soon as possible, as in the case for traffic condition alerts. Figure 5.8(c) illustrates that Push-and-Track could transmit slightly larger content than GO-DCR and Linger, since contacts among replicas and target vehicles last longer. However, this accomplishment is undervalued due to the lower coverage obtained.

GO-DCR also performed better than Push-and-Track when it comes to delivery cost in terms of the number of V2I messages exchanged, as shown in Figure 5.9(a). Given this result, we argue that the objective of giving preference to less expensive communication, V2V, was achieved by GO-DCR, as illustrated in Figure 5.9(b). Since Linger does not rely on infrastructure stations, it does not require infrastructure messages to operate. Linger also does not require many V2V messages, a result of the bad replica allocation.

However, the improvement in content availability and the preference for V2V communication come at a price. GO-DCR initiated a greater amount of redundant data than Push-and-Track and Linger because the replica vehicles were in more frequent contact with the same target vehicles, as shown in Figure 5.9(c). Regardless, the redundant data in GO-DCR is a result of V2V content transmissions, which is less critical than using the most expensive V2I communication.

5.3.2.3 Network-enabled study

The objective of this study is to evaluate GO-DCR when vehicular specific network protocols are used.



(a) Coverage (b) Time to be Covered

Figure 5.10. Network-enabled simulation results in terms of content availability.

We adopt a Manhattan-like mobility scenario comprised of vertical and horizontal double-lane roads in a 9 km^2 area, in which blocks have an average size of 80 m × 270 m. Manhattan was chosen because of its similarity with many other cities around the world in terms of roads and traffic. To simulate realistic vehicle movements, we take advantage of the mobility model defined by SUMO[4]. We fixed the radius of the RoI at 500 m to be in accordance with the simulated area. We consider a static content size of 100 Kbytes, as a single fragment of the same size, to be delivered to vehicles that are inside the RoI during the entire application running time. We set the other parameters, such as delivery interval and content size, with the same values as in the large-scale evaluation study. We run simulations in low, medium, and high network density to evaluate different scenarios.

The network-enabled content availability results are shown in Figure 5.10. Given the confidence interval, all solutions achieved 100% coverage for all network densities,

---

[4] http://sumo-sim.org

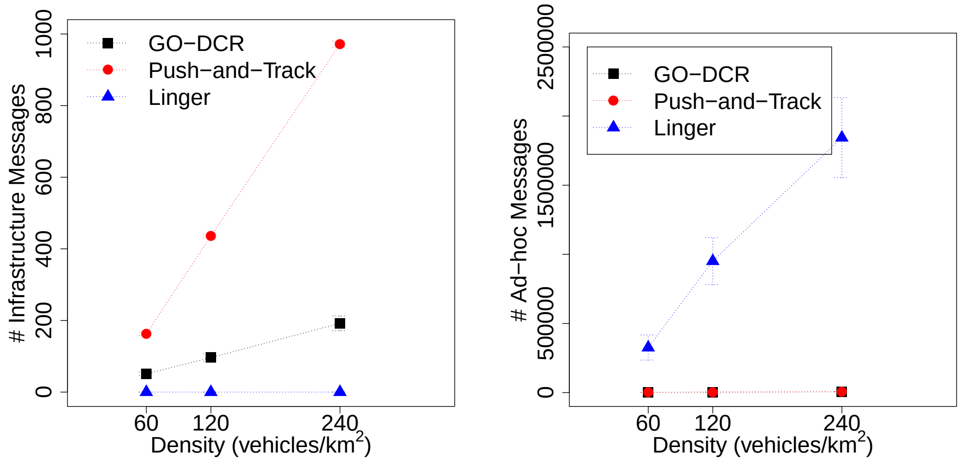
as shown in Figure 5.10(a). This result was expected due to the size of the network scenario, which increases the chance of a target vehicle coming into contact with a replica vehicle. When it comes to the time-to-be-covered, Linger and Push-and-Track present slightly better values, particularly for low network densities, as depicted in Figure 5.10(b). This is due to the fact that GO-DCR does not focus on covering vehicles as soon as possible, but on balancing the number of replicas along time and space. As a result, target vehicles at the beginning of the content lifetime may not be in immediate contact with a replica vehicle. In addition, Push-and-Track selects more replicas than GO-DCR, which increases the chance of target vehicles making contact with replica vehicles. In fact, we measured that Push-and-Track selects on average approximately 3.7 times more replica vehicles than GO-DCR. Linger also performs quite well on this metric, regardless of the network density. For such a small-scale scenario, the distributed index-based approach followed by Linger is very useful. However, when we also consider the large-scale evaluation results, GO-DCR seems more attractive. Furthermore, these results come at a price for Push-and-Track, as shown by the communication cost results in Figure 5.11.

With respect to communication costs, Figure 5.11(a) shows that Push-and-Track requires a significantly higher number of V2I messages to be exchanged during its operation, when compared to GO-DCR and Linger. In fact, this is a key point, since V2I communication is more expensive than V2V. Again, Linger requires no V2I messages, since it relies exclusively on V2V communication.

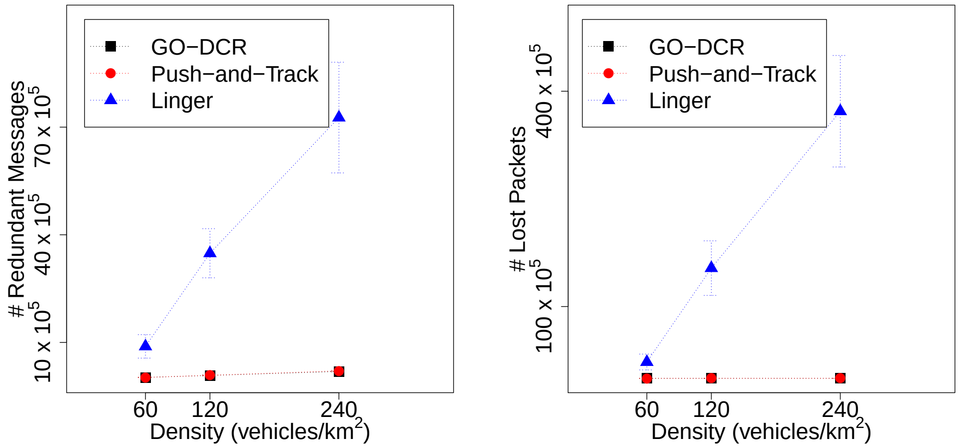
On the other hand, both Push-and-Track and GO-DCR present similar results for V2V communication costs and redundancy, with slightly better values for GO-DRC, as shown in Figures 5.11(b) and 5.11(c). These results are mainly because Push-and-Track selects more replica vehicles (i.e., approximately 3.7 times on average) and adopts the panic zone approach using V2I communication. On the other hand, GO-DCR balances the number of replicas over time to give more opportunities for using V2V communication. Linger, on the other hand, requires a larger number of V2V, and consequently, generates more redundant messages than the other solutions. Furthermore, these values increase in a manner linear to with the network density, since Linger requires more negotiation messages in the replica selection process.

Finally, Figure 5.11(d) shows that Linger also results in greater packet loss, mainly due to communication congestion caused by the large number of messages required. This is even more critical for scenarios with greater network density.

It is important to discuss differences observed when comparing large-scale with network-enabled results. For example, Linger performed quite well in the small-scale, network-enabled scenario, and very poorly in the large-scale scenario. In other words,



(a) Cost – V2I Messages (b) Cost - V2V Messages



(c) Redundancy (d) Lost Packets

Figure 5.11. Network-enabled Simulation Results in Terms of Cost (V2I Messages, V2V Messages, Redundancy and Network Lost Packets)

Linger, as originally proposed, should be used in small scenarios and avoided in large-scale ones. Therefore, we argue that each solution has its advantages and disadvantages.

In general, considering both large-scale and network-enabled results, we can state that our proposal, GO-DCR, achieves high coverage results and conserves resources by requiring a low number of V2I exchanged messages. We accomplished our objective of balancing the number of replicas over time, covering as many vehicles as possible and using as little infrastructure as possible. In conclusion, GO-DCR is a cost-effective solution that could be adopted in various scenarios.

5.4 Final Remarks

In this chapter, we implement the proposed VCDN model for two different applications: city-wide and region-wide. In the first, the content should be delivered to all vehicles in the network. On the other hand, in the second the content should be delivered only to vehicles that are inside a region of interest during the content lifetime. Both solutions rely on the characterization results in terms of O-D points, and the results reveal that they were able to improve the content availability while reducing the communication cost. In the next chapter, we present our conclusions and future directions and the publications obtained as well.

Chapter 6

Final Remarks

6.1 Conclusion

In this thesis, we explored the field of content delivery for vehicular ad hoc networks (VANETs), with the hypothesis that concepts inherited from Content Delivery Network (CDN) and Peer-to-Peer (P2P) and adapted to VANET applications would improve their performance. Given that, the objective was to investigate how those concepts could be applied to VANETs, and propose solutions that validate the aforementioned hypothesis.

To accomplish this objective, we first did an in-depth study of the state-of-the-art to understand how content delivery has been explored in the context of VANETs. In this study, we realized that there was a lot of room for improvements in this field, particularly when it comes to content delivery models and frameworks. Therefore, we proposed a Vehicular Content Delivery Network (VCDN) model and a framework to guide designers when developing VANET applications with content delivery demand. We proposed a hybrid model that inherits concepts from both traditional CDN and P2P, adapting them to VANETs' concept.

To validate our model, we implemented two solutions for different scenarios: city-wide and region-wide. These solutions were proposed based on characterization results obtained within this thesis' context, which indicates that origin-destination (O-D) points seem to be useful for content replication and delivery. We conducted extensive simulations under large-scale scenarios to demonstrate that our VCND model was very effective for content delivery in VANETs. In addition to validate our hypothesis, these solutions also advance the state-of-the-art as they explore novel aspects to select good replica vehicles.

6.2 Future Directions

We discuss here some important research opportunities to extend this thesis:

- Pull-based applications: In addition to the push-based proposals described in this thesis, it is also important to evaluate how our VCDN model will perform under pull-based applications where vehicles send explicit request messages and, therefore, the content discovery is even more challenging;

- Real-time content: A promising type of application for VANETs involves the transmission of real-time videos to be consumed on-board. The delay-sensitive characteristic of such content turns the replication and delivering tasks even more difficult. Thus, an important future research direction is related to proposing and evaluating solutions with real-time content demand;

- Dynamic content: The highly dynamic nature of VANETs also leads to constant changes in content to be delivered to on-board users. Therefore, the content management module should be implemented and evaluated under various conditions for many applications;

- Network-layer integration: As stated earlier, our VCDN model is compliant with different lower-level solutions. However, it is important to evaluate how to integrate our model with different network-level protocols, particularly the ones that follow the Information Centric Network (ICN) concepts;

- Incentive mechanisms: Cooperation will play a determining role on VANET content delivery applications. Thus, novel incentive mechanisms that benefit selfless and punish selfish users should be proposed by taken into account the particular characteristics of VANETs;

- Mobility Scenarios: Given the diversity of cities all over the world, it is also important to evaluate our VCDN model for different mobility scenarios, including highway and urban other than Cologne, Manhattan and Ottawa.

- Interactive applications: Other promising VANET applications involve interactive content, such as online gaming. In this case, users interact with each other in real-time, bringing even more difficult to content replication and delivery. This type of application should also be explored in the near future.

6.3 Publications

During the development of this work, we have produced and published the following papers:

6.3.1 Journal Papers

1. **Fabrício A. Silva**; Azzedine Boukerche; Thais R. M. Braga Silva; Linnyer B. Ruiz; Eduardo Cerqueira; Antonio A. F. Loureiro, "Vehicular Networks: A New Challenge for Content Delivery-based Applications". *ACM Computing Surveys*

2. **Fabrício A. Silva**; Azzedine Boukerche; Thais R. M. Braga Silva; Linnyer B. Ruiz; Antonio A. F. Loureiro, "Geo-localized Content Availability in VANETs". *Ad Hoc Networks (Elsevier)*, 36(2):425-434 , January 2016.

3. **Fabrício A. Silva**; Azzedine Boukerche; Thais R. M. Braga Silva; Fabrício Benevenuto; Linnyer B. Ruiz; Antonio A. F. Loureiro, "ODCRep: Origin-Destination-based Content Replication for Vehicular Networks". *IEEE Transactions on Vehicular Technology* 64(12):1-12, December 2015.

4. **Fabrício A. Silva**; Azzedine Boukerche; Thais R. M. Braga Silva; Linnyer B. Ruiz; Antonio A. F. Loureiro, "A novel macroscopic mobility model for vehicular networks". *Computer Networks (Elsevier)*, 79(C):188-202, March 2015.

6.3.2 Conference Papers

1. **Fabrício A. Silva**; Clayson Celes; Azzedine Boukerche; Linnyer B. Ruiz; Antonio A. F. Loureiro, "Filling the Gaps of Vehicular Mobility Traces". *ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*,(2015).

2. **Fabrício A. Silva**; Azzedine Boukerche; Thais R. M. Braga Silva; Linnyer B. Ruiz; Antonio A. F. Loureiro, "Geo-localized Content Replication for Vehicular Ad-hoc Networks". *IEEE Symposium on Computers and Communications (ISCC)*,(2015):695-700.

3. **Fabrício A. Silva**; Linnyer B. Ruiz; Antonio A. F. Loureiro, "Content Replication in Mobile Vehicular Ad-hoc Networks". *IEEE Mobile Data Management (MDM)*,(2015):26-29.

4. **Fabrício A. Silva**; Thais R. M. Braga Silva; Fabrício Benevenuto; Linnyer B. Ruiz; Antonio A. F. Loureiro, "Improving information dissemination in vehicular networks by selecting appropriate disseminators". *28th IEEE International Conference on Advanced Information Networking and Applications (AINA)*,(2014):681-688.

5. **Fabrício A. Silva**; Azzedine Boukerche; Thais R. M. Braga Silva; Linnyer B. Ruiz; Eduardo Cerqueira; Antonio A. F. Loureiro, "Content Replication and Delivery in Vehicular Networks". *4th ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications (DIVANet)*, (2014):127-132.

6. Ronan D. Mendonça; Thais R. M. Braga Silva; **Fabrício A. Silva**; Linnyer B. Ruiz; Antonio A. F. Loureiro, "Dynamic Bandwidth Distribution for Entertainment Vehicular Networks Applications". *International Workshop on Pervasive Internet of Things and Smart Cities (PITSaC)*, (2014):827-832.

7. **Fabrício A. Silva**; Thais R. M. Braga Silva; Linnyer B. Ruiz; Antonio A. F. Loureiro; Rafael Vicente, "On the improvement of vehicular macroscopic mobility models". *IEEE 78th Vehicular Technology Conference (VTC-Fall)*, (2013):1-5.

8. **Fabrício A. Silva**; Thais R. M. Braga Silva; Linnyer B. Ruiz; Antonio A. F. Loureiro, "ConProVA: A Smart Context Provisioning Middleware for VANET Applications". *IEEE 77th Vehicular Technology Conference (VTC-Spring)*, (2013):1-5.

9. **Fabrício A. Silva**; Thais R. M. Braga Silva; Linnyer B. Ruiz; Antonio A. F. Loureiro, "Um Middleware para provisionamento de contextos para redes veiculares". *Simpósio Brasileiro de Redes de Computadores (SBRC)*, (2013):615-628.

Bibliography

(2015). Waze. Online (available at http://www.waze.com).

Acer, U., Giaccone, P., Hay, D., Neglia, G., and Tarapiah, S. (2011). Timely data delivery in a realistic bus network. In *International Conference on Computer Communications (INFOCOM 2011)*, pages 446--450, Shanghai, China. IEEE.

Ahlgren, B., Dannewitz, C., Imbrenda, C., Kutscher, D., and Ohlman, B. (2012). A survey of information-centric networking. *IEEE Communications Magazine*, 50(7):26--36.

Ahmed, S. and Kanhere, S. S. (2006). Vanetcode: Network coding to enhance cooperative downloading in vehicular ad-hoc networks. In *International Conference on Communications and Mobile Computing (IWCMC)*, pages 527--532. ACM.

Ahmed, S. and Salil, S. K. (2010). Characterization of a large-scale delay tolerant network. In *IEEE 35th Conference on Local Computer Networks (LCN)*, pages 56--63, Denver, CO.

Amadeo, M., Campolo, C., and Molinaro, A. (2012). CRoWN: Content-Centric Networking in Vehicular Ad Hoc Networks. *IEEE Communications Letters*, 16(9):1380--1383. ISSN 1089-7798.

Amadeo, M., Campolo, C., and Molinaro, A. (2013). Enhancing content-centric networking for vehicular environments. *Computer Networks*, 57(16):3222--3234. ISSN 1389-1286.

Androutsellis-Theotokis, S. and Spinellis, D. (2004). A survey of peer-to-peer content distribution technologies. *ACM Computing Surveys (CSUR)*, 36(4):335--371.

Bai, F. and Krishnamachari, B. (2010). Exploiting the wisdom of the crowd: localized, distributed information-centric VANETs. *IEEE Communications Magazine*, 48(5):138--146. ISSN 0163-6804.

Bai, F., Krishnan, H., Elbatt, T., and Holland, G. (2012). Towards characterising and classifying communication-based automotive applications from a wireless networking perspective. *International Journal of Vehicle Autonomous Systems*, 10(3):165--197.

Baier, P., Durr, F., and Rothermel, K. (2012). TOMP: Opportunistic traffic offloading using movement predictions. In *Conference on Local Computer Networks*, pages 50--58, Clearwater, Florida, USA. IEEE.

Baiocchi, A. and Cuomo, F. (2013). Infotainment services based on push-mode dissemination in an integrated VANET and 3G architecture. *Journal of Communications and Networks*, 15(2):179--190. ISSN 1229-2370.

Barbera, M. V., Viana, A. C., De Amorim, M. D., and Stefa, J. (2014). Data offloading in social mobile networks through vip delegation. *Ad Hoc Networks*, 19:92--110.

Barros, J. (2014). How to build vehicular networks in the real world. In *Proceedings of the 15th ACM international symposium on Mobile ad hoc networking and computing*, pages 123--124. ACM.

Blair, G., Coulouris, G., Dollimore, J., and Kindberg, T. (2012). Distributed systems: Concepts and design.

Borgia, E. (2014). The internet of things vision: Key features, applications and open issues. *Computer Communications*, 54:1--31.

Borsetti, D., Fiore, M., Casetti, C., and Chiasserini, C.-F. (2011). An application-level framework for information dissemination and collection in vehicular networks. *Performance Evaluation*, 68(9):876--896. ISSN 01665316.

Bruno, F., Cesana, M., Gerla, M., Mauri, G., and Verticale, G. (2014). Optimal content placement in icn vehicular networks. In *The 5th International Conference On Network of the Future (NOF 2014)*, Paris, France. IEEE.

Bruno, F., Cesana, M., Gerla, M., Mauri, G., and Verticale, G. (2015). Optimal content placement in icn vehicular networks. In *5th IEEE International Conference on Network of the Future.*

Bujari, A. (2012). A Survey of Opportunistic Data Gathering and Dissemination Techniques. In *International Conference on Computer Communications and Networks (ICCCN)*, pages 1--6, Munich, Germany. IEEE.

Cao, Y., Guo, J., and Wu, Y. (2014). Sdn enabled content distribution in vehicular networks. In *International Conference on Innovative Computing Technology (INTECH)*, pages 164--169. IEEE.

Carlsson, B. and Gustavsson, R. (2001). The rise and fall of napster-an evolutionary approach. In *Active Media Technology*, pages 347--354. Springer.

Carofiglio, G., Morabito, G., Muscariello, L., Solis, I., and Varvello, M. (2013). From content delivery today to information centric networking. *Computer Networks*, 57(16):3116--3127.

Casella, G. and Berger, R. (2001). *Statistical Inference.* Duxbury Resource Center, 2nd edition.

Chen, B. B. and Chan, M. C. (2009). MobTorrent: A Framework for Mobile Internet Access from Vehicles. In *Conference on Computer Communications*, pages 1404--1412. IEEE.

Cheng, L., Henty, B., Stancil, D., Bai, F., and Mudalige, P. (2007). Mobile vehicle-to-vehicle narrow-band channel measurement and characterization of the 5.9 ghz dedicated short range communication (dsrc) frequency band. *IEEE Journal on Selected Areas in Communications*, 25(8):1501--1516.

Ciccarese, G., De Blasi, M., Marra, P., Palazzo, C., and Patrono, L. (2009). On the Use of Control Packets for Intelligent Flooding in VANETs. In *Wireless Communications and Networking Conference (WCNC)*, pages 1--6, Budapest. IEEE.

Cohen, B. (2003). Incentives build robustness in bittorrent. In *Workshop on Economics of Peer-to-Peer systems*, volume 6, pages 68--72.

Costa-Montenegro, E., Quiñoy-García, F., González-Castaño, F., and Gil-Castineira, F. (2012). Vehicular entertainment systems: Mobile application enhancement in networked infrastructures. *IEEE Vehicular Technology Magazine*, 7(3):73–79. ISSN 1556-6072.

Department, I. S. A. (2010). Ieee 802.11p/d10.0.

Derhab, A. and Badache, N. (2006). Localized Hybrid Data Delivery Scheme using K-hop Clustering Algorithm in Ad Hoc Networks. In *International Conference on Mobile Ad Hoc and Sensor Sysetems*, pages 668--673. IEEE.

Eugster, P. T., Felber, P. A., Guerraoui, R., and Kermarrec, A.-M. (2003). The many faces of publish/subscribe. *ACM Computing Surveys (CSUR)*, 35(2):114--131.

Fiore, M., Casetti, C., and Chiasserini, C.-F. (2005). On-demand content delivery in vehicular wireless networks. In *International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, pages 87--94. ACM.

Fiore, M., Casetti, C., and Chiasserini, C.-f. (2011). Caching Strategies Based on Information Density Estimation in Wireless Ad Hoc Networks. *IEEE Transactions on Vehicular Technology*, 60(5):2194--2208. ISSN 0018-9545.

Fiore, M., Casetti, C., Chiasserini, C.-f., and Borsetti, D. (2013). Persistent Localized Broadcasting in VANETs. *IEEE Journal on Selected Areas in Communications*, 31(9):480--490. ISSN 0733-8716.

Fiore, M., Mininni, F., Casetti, C., and Chiasserini, C.-F. (2009). To Cache or Not To Cache? In *INFOCOM*, pages 235--243. IEEE.

Gerla, M., Lee, E.-K., Pau, G., and Lee, U. (2014a). Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds. In *World Forum on Internet of Things (WF-IoT)*, pages 241--246. IEEE.

Gerla, M., Wu, C., Pau, G., and Zhu, X. (2014b). Content distribution in VANETs. *Vehicular Communications*, 1(1):3--12. ISSN 2214-2096.

Gkantsidis, C. and Rodriguez, P. R. (2005). Network coding for large scale content distribution. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 4, pages 2235--2245. IEEE.

Grassi, G., Pesavento, D., Pau, G., Vuyyuru, R., Wakikawa, R., and Zhang, L. (2014). Vanet via named data networking. In *Conference on Computer Communications Workshops (INFOCOM WORKSHOPS)*, pages 410--415. IEEE.

Guidec, F. and Maheo, Y. (2007). Opportunistic Content-Based Dissemination in Disconnected Mobile Ad Hoc Networks. In *International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM)*, pages 49--54. IEEE.

Ha, D. and Ngo, H. (2009). MoPADS: A Mobility Profile Aided File Downloading Service in Vehicular Networks. *IEEE Transactions on Vehicular Technology*, 58(9):5235--5246. ISSN 0018-9545.

Han, B., Hui, P., Kumar, V. A., Marathe, M. V., Pei, G., and Srinivasan, A. (2010). Cellular traffic offloading through opportunistic communications: a case study. In *Workshop on Challenged Networks*, pages 31--38. ACM.

Han, B., Hui, P., Kumar, V. A., Marathe, M. V., Shao, J., and Srinivasan, A. (2012). Mobile Data Offloading through Opportunistic Communications and Social Participation. *IEEE Transactions on Mobile Computing*, 11(5):821--834. ISSN 1536-1233.

He, K., Li, X., Schick, B., Qiao, C., Sudhaakar, R., Addepalli, S., and Chen, X. (2013). On-road video delivery with integrated heterogeneous wireless networks. *Ad Hoc Networks*, 11(7):1992--2001. ISSN 15708705.

Hofmann, M. and Beaumont, L. R. (2005). *Content networking: architecture, protocols, and practice.* Elsevier.

Huang, J.-L., Chen, M.-S., and Peng, W.-c. (2003). Exploring group mobility for replica data allocation in a mobile environment. In *International Conference on Information and Knowledge Management (CIKM)*, pages 161--168. ACM.

Jeong, J., Guo, S., Gu, Y., He, T., and Du, D. H. (2010). TSF: Trajectory-Based Statistical Forwarding for Infrastructure-to-Vehicle Data Delivery in Vehicular Networks. In *International Conference on Distributed Computing Systems*, pages 557--566. IEEE.

Jiang, D. and Delgrossi, L. (2008). Ieee 802.11p: Towards an international standard for wireless access in vehicular environments. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, pages 2036–2040. ISSN 1550-2252.

Joerer, S., Dressler, F., and Sommer, C. (2012). Comparing apples and oranges? In *ACM International Workshop on Vehicular Inter-networking, Systems and Applications*, pages 27--32, New York, USA.

Karagiannis, G., Altintas, O., Ekici, E., Heijenk, G., Jarupan, B., Lin, K., and Weil, T. (2011). Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions. *Communications Surveys & Tutorials*, 13(4):584--616.

Katti, S., Rahul, H., Hu, W., Katabi, D., Médard, M., and Crowcroft, J. (2008). Xors in the air: Practical wireless network coding. *IEEE/ACM Trans. Netw.*, 16(3):497--510. ISSN 1063-6692.

Kawadia, V., Riga, N., and Opper, J. (2011). Slinky : An Adaptive Protocol for Content Access in Disruption-Tolerant Ad Hoc Networks. In *MobiHoc Workshop on Tactical Mobile Ad Hoc Networking.*

Khabbaz, M. J., Fawaz, W. F., and Assi, C. M. (2012). Which Vehicle To Select? *IEEE Communications Letters*, 16(6):812--815. ISSN 1089-7798.

Khaitiyakun, N. and Sanguankotchakorn, T. (2014). An analysis of data dissemination on vanet by using content delivery network (cdn) technique. In *Asian Internet Engineering Conference (AINTEC)*, page 37. ACM.

Kumar, N., Rodrigues, J. J., Lloret, J., and Dua, A. (2015). Replication-aware data dissemination for vehicular ad hoc networks using location determination. *Mobile Networks and Applications*, pages 1--17.

La, C.-a., Michiardi, P., Casetti, C., Chiasserini, C.-f., and Fiore, M. (2012). Content Replication in Mobile Networks. *IEEE Journal on Selected Areas in Communications*, 30(9):1762--1770. ISSN 0733-8716.

Law, A. M. and Kelton, D. M. (1999). *Simulation Modeling and Analysis.* McGraw-Hill Higher Education, 3rd edition.

Lee, E., Lee, E.-k., and Gerla, M. (2014). Vehicular Cloud Networking: Architecture and Design Principles. *IEEE Communications Magazine*, (2):148--155.

Lee, K. C., Lee, S.-H., Cheung, R., Lee, U., and Gerla, M. (2007). First experience with cartorrent in a real vehicular ad hoc network testbed. In *IEEE Mobile Networking for Vehicular Environments*, pages 109--114.

Leontiadis, I. (2007). Publish/subscribe notification middleware for vehicular networks. In *Middleware Doctoral Symposium*, pages 1--6. ACM.

Leontiadis, I., Costa, P., and Mascolo, C. (2009a). A hybrid approach for content-based publish/subscribe in vehicular networks. *Pervasive and Mobile Computing*, 5(6):697--713. ISSN 1574-1192.

Leontiadis, I., Costa, P., and Mascolo, C. (2009b). Persistent content-based information dissemination in hybrid vehicular networks. In *International Conference on Pervasive Computing and Communications*, pages 1--10. IEEE.

Leontiadis, I., Costa, P., and Mascolo, C. (2010). Extending Access Point Connectivity through Opportunistic Routing in Vehicular Networks. In *INFOCOM*, pages 1--5. IEEE.

Leontiadis, I. and Mascolo, C. (2007). Opportunistic spatio-temporal dissemination system for vehicular networks. In *International MobiSys Workshop on Mobile Opportunistic Networking*, pages 39--46. ACM.

Li, F. and Wang, Y. (2007). Routing in vehicular ad hoc networks: A survey. *IEEE Vehicular Technology Magazine*, 2(2):12--22.

Li, Y. J. (2012). An overview of the dsrc/wave technology. In *Quality, Reliability, Security and Robustness in Heterogeneous Networks*, pages 544--558. Springer.

Lin, K. C.-j., Chen, C.-w., and Chou, C.-f. (2012). Preference-aware content dissemination in opportunistic mobile social networks. In *INFOCOM*, pages 1960--1968. IEEE.

Liu, H., Chen, Z., Tian, X., Wang, X., and Tao, M. (2014). On content-centric wireless delivery networks. *IEEE Wireless Communications*, 21(6):118--125.

Lu, N., Cheng, N., Zhang, N., Shen, X., and Mark, J. W. (2014). Connected vehicles: Solutions and challenges. *IEEE Internet of Things Journal*, 1(4):289--299.

Lu, S., Chavan, G., Liu, Y., and Liu, Y. (2011). Design and Analysis of a Mobile File Sharing System for Opportunistic Networks. In *International Conference on Computer Communications and Networks (ICCCN)*, pages 1--6. IEEE.

Lua, E. K., Crowcroft, J., Pias, M., Sharma, R., and Lim, S. (2005). A survey and comparison of peer-to-peer overlay network schemes. *Communications Surveys & Tutorials, IEEE*, 7(2):72--93.

Luan, T., Cai, L., Chen, J., Shen, X., and Bai, F. (2014). Engineering a distributed infrastructure for large-scale cost-effective content dissemination over urban vehicular networks. *Vehicular Technology, IEEE Transactions on*, 63(3):1419–1435. ISSN 0018-9545.

Luan, T. H., Cai, L. X., Chen, J., Shen, X., and Bai, F. (2011). Vtube: Towards the media rich city life with autonomous vehicular content distribution. In *Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, pages 359--367. IEEE.

Maia, G., Rezende, C., Villas, L. A., Boukerche, A., Viana, A. C., Aquino, A. L., and Loureiro, A. A. (2013). Traffic aware video dissemination over vehicular ad hoc networks. In *International conference on Modeling, analysis & simulation of wireless and mobile systems*, pages 419--426. ACM.

Maihöfer, C., Leinmüller, T., and Schoch, E. (2005). Abiding Geocast: Time–stable Geocast for Ad Hoc Networks. In *International Workshop on Vehicular Ad Hoc Networks*, pages 20--29. ACM.

Malandrino, F., Casetti, C., and Chiasserini, C.-f. (2012). Content Discovery and Caching in Mobile Networks with Infrastructure. *IEEE Transactions on Computers*, 61(10):1507--1520. ISSN 0018-9340.

Malandrino, F., Casetti, C., Chiasserini, C.-f., and Fiore, M. (2013). Optimal Content Downloading in Vehicular Networks. *IEEE Transactions on Mobile Computing*, 12(7):1377--1391. ISSN 1536-1233.

Malandrino, F., Casetti, C., Chiasserini, C.-F., and Fiore, M. (2014). Content download in vehicular networks in presence of noisy mobility prediction. *Mobile Computing, IEEE Transactions on*, 13(5):1007--1021.

Martin, P. and Hassanein, H. (2005). Cluster-based Replication for Large-scale Mobile Ad-hoc Networks. In *International Conference on Wireless Networks, Communications and Mobile Computing*, volume 1, pages 552--557. IEEE.

Mislove, A., Marcon, M., Gummadi, K. P., Druschel, P., and Bhattacharjee, B. (2007). Measurement and analysis of online social networks. In *7th ACM SIGCOMM Conference on Internet Measurement*, pages 29--42, New York, NY, USA.

Monteiro, R., Sargento, S., Viriyasitavat, W., and Tonguz, O. K. (2012). Improving VANET protocols via network science. In *2012 IEEE Vehicular Networking Conference (VNC)*, pages 17--24. IEEE.

Morgan, Y. L. (2010). Notes on dsrc & wave standards suite: Its architecture, design, and characteristics. *Communications Surveys & Tutorials, IEEE*, 12(4):504--518.

Nadeem, T., Shankar, P., and Iftode, L. (2006). A Comparative Study of Data Dissemination Models for VANETs. In *Workshops of International Conference on Mobile and Ubiquitous Systems*, pages 1--10.

Nandan, A., Das, S., Pau, G., Gerla, M., and Sanadidi, M. (2005). Co-operative Downloading in Vehicular Ad-Hoc Wireless Networks. In *Annual Conference on Wireless On-demand Network Systems and Services*, pages 32--41. IEEE.

Naumov, V., Baumann, R., and Gross, T. (2006). An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces. In *The 7th ACM International*

Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), pages 108--119, New York, NY, USA.

Padmanabhan, P., Gruenwald, L., Vallur, A., and Atiquzzaman, M. (2008). A survey of data replication techniques for mobile ad hoc network databases. *The International Journal on Very Large Data Bases*, 17(5):1143--1164.

Pallis, G. and Vakali, A. (2006). Insight and perspectives for content delivery networks. *Communications of the ACM*, 49(1):101--106.

Parameswaran, M., Susarla, A., and Whinston, A. B. (2001). P2p networking: An information-sharing alternative. *Computer*, 34(7):31–38. ISSN 0018-9162.

Passarella, A. (2012). A survey on content-centric technologies for the current internet: Cdn and p2p solutions. *Computer Communications*, 35(1):1--32.

Pathan, A. and Buyya, R. (2007). A Taxonomy and Survey of Content Delivery Networks. Technical report, Grid Computing and Distributed Systems Laboratory, Melbourne.

Peng, G. (2004). CDN: Content Distribution Network. *arXiv preprint cs/0411069*.

Piro, G., Cianci, I., Grieco, L. A., Boggia, G., and Camarda, P. (2014). Information centric services in smart cities. *Journal of Systems and Software*, 88:169--188.

Plackett, R. L. (1983). Karl pearson and the chi-squared test. *International Statistical Review*, 51(1):59--72.

Ramazani, A. and Vahdat-Nejad, H. (2014). A new context-aware approach to traffic congestion estimation. In *4th International eConference on Computer and Knowledge Engineering (ICCKE)*, pages 504--508. IEEE.

Resta, G. and Santi, P. (2010). On the Fundamental Limits of Broadcasting in Wireless Mobile Networks. In *INFOCOM*, pages 1--5, San Diego, CA. IEEE.

Rezende, C., Mammeri, A., Boukerche, A., and Loureiro, A. A. (2014). A receiver-based video dissemination solution for vehicular networks with content transmissions decoupled from relay node selection. *Ad Hoc Networks*, 17(June):1--17. ISSN 1570-8705.

Rezende, C., Ramos, H. S., Pazzi, R. W., Boukerche, A., Frery, A. C., and Loureiro, A. A. F. (2012). Virtus: A resilient location-aware video unicast scheme for vehicular networks. In *International Conference on Communications*, pages 698--702. IEEE.

Rostamzadeh, K. and Gopalakrishnan, S. (2011). Analysis of emergency message dissemination in vehicular networks. In *Wireless Communications and Networking Conference*, pages 575--580, Cancun, Quintana Roo. IEEE.

Ruiz, P., Dorronsoro, B., Bouvry, P., and Tardón, L. (2012). Information dissemination in VANETs based upon a tree topology. *Ad Hoc Networks*, 10(1):111--127. ISSN 15708705.

Sanguesa, J. A., Fogue, M., Garrido, P., Martinez, F. J., Cano, J.-C., Calafate, C. T., and Manzoni, P. (2015). Rtad: A real-time adaptive dissemination system for vanets. *Computer Communications*. ISSN 0140-3664.

Sardari, M., Hendessi, F., and Fekri, F. (2009). Infocast: A New Paradigm for Collaborative Content Distribution from Roadside Units to Vehicular Networks. In *Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pages 1--9. IEEE.

Sharma, P., Kant, K., and Chauhan, N. (2010). A Comparative Study of Cluster-based Data Replication Techniques for MANETs. *International Journal of Information Technology*, 2(2):665--667.

Sommer, C., Eckhoff, D., German, R., and Dressler, F. (2010). A Computationally Inexpensive Empirical Model of IEEE 802.11p Radio Shadowing in Urban Environments. Technical Report CS-2010-06, University of Erlangen, Dept. of Computer Science.

Stanica, R., Fiore, M., and Malandrino, F. (2013). Offloading Floating Car Data. In *International Symposium on A World of Wireless, Mobile and Multimedia Networks"*, pages 1--9. IEEE.

Stiemerling, M. and Kiesel, S. (2009). A system for peer-to-peer video streaming in resource constrained mobile environments. In *Workshop on User-provided networking: challenges and opportunities*, pages 25--30. ACM.

Stiemerling, M. and Kiesel, S. (2010). Cooperative P2P Video Streaming for Mobile Peers. In *International Conference on Computer Communications and Networks*, pages 1--7. IEEE.

Sung, Y. and Lee, M. (2012). Light-Weight Reliable Broadcast Message Delivery for Vehicular Ad-Hoc Networks. In *Vehicular Technology Conference (VTC Spring)*, pages 1--6, Yokohama. IEEE.

Teixeira, F. A., e Silva, V. F., Leoni, J. L., Macedo, D. F., and Nogueira, J. M. (2014). Vehicular networks using the ieee 802.11 p standard: An experimental analysis. *Vehicular Communications*, 1(2):91--96.

Telefonica (2014). Connected car industry report.

Thilakarathna, K., Seneviratne, A., Viana, A. C., and Petander, H. (2014). User generated content dissemination in mobile social networks through infrastructure supported content replication. *Pervasive and Mobile Computing*, 11:132--147.

Thilakarathna, K., Viana, A. C., Seneviratne, A., and Petander, H. (2013). Mobile social networking through friend-to-friend opportunistic content dissemination. In *Proceedings of the fourteenth ACM international symposium on Mobile ad hoc networking and computing*, pages 263--266. ACM.

Trullols-Cruces, O., Fiore, M., and Barcelo-Ordinas, J. (2012). Cooperative Download in Vehicular Environments. *IEEE Transactions on Mobile Computing*, 11(4):663--678. ISSN 1536-1233.

Uppoor, S. and Fiore, M. (2011). Large-scale urban vehicular mobility for networking research. In *2011 IEEE Vehicular Networking Conference (VNC)*, pages 62--69. IEEE.

Uppoor, S. and Fiore, M. (2012). Insights on metropolitan-scale vehicular mobility from a networking perspective. In *International Workshop on Hot Topics in Planet-scale Measurement*, pages 39--44, Low Wood Bay, Lake District, UK. ACM.

Uppoor, S., Naboulsi, D., and Fiore, M. (2014a). Vehicular mobility trace vehicular mobility trace of the city of cologne. Online (available at http://kolntrace.project.citi-lab.fr).

Uppoor, S., Trullols-Cruces, O., Fiore, M., and Barcelo-Ordinas, J. M. (2014b). Generation and Analysis of a Large-Scale Urban Vehicular Mobility Dataset. *IEEE Transactions on Mobile Computing*, 13(5). ISSN 1536-1233.

Vahdat, A., Becker, D., et al. (2000). Epidemic routing for partially connected ad hoc networks. Technical report, Technical Report CS-200006, Duke University.

Vakali, A. and Pallis, G. (2003). Content delivery networks: Status and trends. *Internet Computing, IEEE*, 7(6):68--74.

Villas, L. A., Boukerche, A., Araujo, R. B., Loureiro, A. A., and Ueyama, J. (2013). Network partition-aware geographical data dissemination. In *International Conference on Communications*, pages 1439--1443. IEEE.

Viriyasitavat, W., Tonguz, O., and Bai, F. (2011). UV-CAST: an urban vehicular broadcast protocol. *IEEE Communications Magazine*, 49(11):116--124. ISSN 0163-6804.

Wang, J., Wakikawa, R., and Zhang, L. (2010). DMND: Collecting data from mobiles using Named Data. In *Vehicular Networking Conference*, pages 49--56. IEEE.

Wang, Y. and Li, F. (2009). Vehicular ad hoc networks. In *Guide to wireless ad hoc networks*, pages 503--525. Springer.

Whitbeck, J., Lopez, Y., Leguay, J., Conan, V., and de Amorim, M. D. (2012). Push-and-track: Saving infrastructure bandwidth through opportunistic forwarding. *Pervasive and Mobile Computing*, 8(5):682--697. ISSN 15741192.

Willke, T., Tientrakool, P., and Maxemchuk, N. (2009). A survey of inter-vehicle communication protocols and their applications. *IEEE Communications Surveys & Tutorials*, 11(2):3--20. ISSN 1553-877X.

Xia, C., Liang, D., Wang, H., Luo, M., and Lv, W. (2012). Characterization and modeling in large-scale urban DTNs. In *IEEE 37th Conference on Local Computer Networks (LCN)*, pages 352--359.

Xu, F., Guo, S., Jeong, J., Gu, Y., Cao, Q., Liu, M., and He, T. (2011). Utilizing shared vehicle trajectories for data forwarding in vehicular networks. In *INFOCOM*, pages 441--445. IEEE.

Younes, M. B. and Boukerche, A. (2015). A performance evaluation of an efficient traffic congestion detection protocol (ecode) for intelligent transportation systems. *Ad Hoc Networks*, 24:317--336.

Younes, M. B., Boukerche, A., and Rom'an-Alonso, G. (2014). An intelligent path recommendation protocol (icod) for vanets. *Computer Networks*, 64:225--242.

Yousefi, S., Mousavi, M. S., and Fathy, M. (2006). Vehicular ad hoc networks (vanets): challenges and perspectives. In *ITS Telecommunications Proceedings, 2006 6th International Conference on*, pages 761--766. IEEE.

Zeadally, S., Hunt, R., Chen, Y.-S., Irwin, A., and Hassan, A. (2012). Vehicular ad hoc networks (VANETS): status, results, and challenges. *Telecommunication Systems*, 50(4):217--241. ISSN 1018-4864.

Zhao, J. and Cao, G. (2008). Vadd: Vehicle-assisted data delivery in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology*, 57(3):1910--1922.

Zhao, J., Member, S., Zhang, Y., Cao, G., and Member, S. (2007). Data Pouring and Buffering on the Road: A New Data Dissemination Paradigm for Vehicular Ad Hoc Networks. *IEEE Transactions on Vehicular Technology*, 56(6):3266--3277. ISSN 0018-9545.

Zyba, G., Voelker, G. M., Ioannidis, S., and Diot, C. (2011). Dissemination in opportunistic mobile ad-hoc networks: The power of the crowd. In *INFOCOM*, pages 1179--1187, Shanghai. IEEE.