DEPLOYMENT OF ROADSIDE UNITS BASED ON PARTIAL MOBILITY INFORMATION

CRISTIANO MACIEL DA SILVA

DEPLOYMENT OF ROADSIDE UNITS BASED ON PARTIAL MOBILITY INFORMATION

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

ORIENTADOR: WAGNER MEIRA JR.

Belo Horizonte - Minas Gerais

16 de dezembro de 2014

CRISTIANO MACIEL DA SILVA

DEPLOYMENT OF ROADSIDE UNITS BASED ON PARTIAL MOBILITY INFORMATION

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

Advisor: Wagner Meira Jr.

Belo Horizonte - Minas Gerais

December 16, 2014

Maciel da Silva, Cristiano

S586d Deployment of Roadside Units Based on Partial Mobility Information / Cristiano Maciel da Silva. — Belo Horizonte - Minas Gerais, 2014
xxix, 204 f. : il. ; 29cm

Tese (doutorado) — Universidade Federal de Minas Gerais
Orientador: Wagner Meira Jr.

1. Computação - Teses. Engenharia de software – Teses. Trânsito – fluxo – Teses. PMCP – Teses.
I. Título.

CDU 519.6*32(043)

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

FOLHA DE APROVAÇÃO

Deployment Of Roadside Units Based On Partial Mobility Information

CRISTIANO MACIEL DA SILVA

Tese defendida e aprovada pela banca examinadora constituída pelos Senhores:

WAGNER MEIRA JÚNIOR
DCC - UFMG

ANTONIO ALFREDO FERREIRA LOUREIRO
DCC - UFMG

EDUARDO COELHO CERQUEIRA
UFPA

JOSÉ MARCOS SILVA NOGUEIRA
DCC - UFMG

RAQUEL APARECIDA DE FREITAS MINI
PUC - MG

Belo Horizonte, 16 de dezembro de 2014.

Resumo

As redes veiculares em breve estarão nas ruas. Dado o papel do automóvel como um componente fundamental do cotidiano, a integração de inteligência de software nos veículos pode drasticamente melhorar a qualidade de vida nos centros urbanos. A adoção de comunicação sem fio nos veículos têm fascinado os pesquisadores desde os anos 80. Alguns anos após, como consequência da revolução celular, os serviços de voz se tornaram amplamente adotados, e a atenção dos pesquisadores voltou-se para os benefícios da comunicação sem fio.

Numa rede veicular a comunicação pode ocorrer de uma forma direta e completamente distribuída onde os veículos trocam mensagens sem nenhuma infraestrutura de suporte. No entanto, a comunicação ad hoc pode se tornar bastante ineficiente devido à grande mobilidade dos veículos que leva a tempos de contatos muito curtos para a troca de dados, possivelmente reduzindo a vazão da rede. Além disso, a comunicação também é dificultada em áreas esparsas devido à ausência de nós comunicantes. A mobilidade veicular também dificulta o roteamento dado que não contamos com meios confiáveis para inferir a posição futura de cada veículo. Embora a comunicação veicular possa se dar de forma ad hoc, diversas pesquisas demonstram que uma infraestrutura mínima de suporte melhora o desempenho geral da rede. Por outro lado, os altos custos de implantação dessa infraestrutura de suporte podem atrasar a implantação em larga escala das redes veiculares. Dessa forma, diversas pesquisas desenvolvem esforços no sentido de maximizar o desempenho da rede através de uma implantação mais eficiente da infraestrutura de apoio.

Estratégias existentes para a implantação de infraestrutura se baseiam em dois paradigmas de mobilidade. As estratégias mais antigas apresentam uma influência clara das redes de telefonia celular, e propõem métodos distintos para a identificação das regiões mais densas da malha viária. Em contrapartida, os trabalhos mais modernos assumem que a trajetória de todos os veículos é conhecida a priori, uma premissa questionável quando consideramos uma implantação de infraestrutura real.

Nesse trabalho nós propomos a implantação de infraestrutura baseada num

modelo parcial de mobilidade. Ao invés de assumirmos o completo conhecimento da trajetória de todos os veículos, nós assumimos o conhecimento das taxas de migrações de veículos entre regiões urbanas adjacentes, uma premissa mais factível. Nós modelamos a alocação de infraestrutura como um Problema de Máxima Cobertura Probabilística (PMCP). PMCP nos permite projetar o fluxo de veículos com o objetivo de identificar a quantidade esperada de veículos atingindo qualquer região da malha viária. Nós usamos essa informação para inferir os melhores locais para a implantação das unidades de comunicação com o objetivo de maximizar a quantidade de veículos distintos contatando a infraestrutura. Nossa estratégia de implantação de infraestrutura é avaliada em três cenários distintos com complexidades crescentes: (a) malha viária sintética com tráfego também sintético; (b) malha viária real com tráfego sintético; (c) malha viária real com tráfego realístico.

Dado que o tráfego varia ao longo do tempo, uma arquitetura baseada somente em unidades de comunicação estacionárias não é capaz de suportar a comunicação veicular durante toda a sua operação. Assim, nós também investigamos os benefícios da utilização de uma infraestrutura dinâmica. Embora o tráfego varie ao longo do tempo, essa variação é de alguma forma limitada pela malha viária subjacente. Assim, nesse trabalho nós também propomos uma arquitetura híbrida baseada na integração de unidades de comunicações móveis e estáticas. Nossa investigação demonstra que aproximadamente 60% das unidades de comunicação podem ser mantidas fixas, enquanto que 40% delas devem se mover ao longo da malha viária. Ao analisarmos as variações de tráfego, nós percebemos que as unidades de comunicações móveis devem se deslocar numa velocidade média de 5.2km/h, o que demonstra a viabilidade de incorporação dessas unidades de comunicação em veículos do transporte público.

Abstract

Vehicular ad-hoc Networks are expected to hit the streets soon. Given the automobile's role as a critical component in peoples' lives, embedding software-based intelligence into them has the potential to drastically improve drivers' quality of life. Leveraging wireless communication in vehicles has fascinated researchers since the 1980s. A few years later, as a consequence of the cellular revolution, voice services have become commonplace and ubiquitous, and the attention of researchers has shifted to wireless communication.

In a vehicular network communication may happen in a direct and completely distributed basis where vehicles exchange messages without any support infrastructure. However, the ad hoc communication may become inefficient due to the high mobility of vehicles leading to very short contact times, possibly reducing the network throughput. Additionally, communication also suffers in sparse areas due to the lack of communicating pairs. Vehicular mobility also makes routing far complicated as we lack reliable means to infer the future position of the vehicles. Although the communication may take place in an ad-hoc basis, the research demonstrates that a minimum support infrastructure may largely improve the overall efficiency of the vehicular network. Nevertheless, the high costs of deploying the support infrastructure may delay the large scale adoption of such networks. Thus, several research efforts are towards maximizing the network performance through more efficient deployment strategies.

Existent deployment proposals rely on two mobility paradigms: Initial deployment works are influenced by the cellular networks, and typically propose alternative strategies to identify the densest places within the road network. On the other hand, modern works assume full knowledge of vehicles trajectories, perhaps a not realistic assumption when we consider a real deployment.

In this work we propose the deployment of infrastructure based on partial mobility information. Instead of assuming full knowledge of the vehicles' trajectories, we assume full knowledge of the migration ratios between adjacent urban locations, a

much more realistic assumption. We model the deployment of roadside units as a Probabilistic Maximum Coverage Problem (PMCP). PMCP allows us to project the vehicles flow in order to identify the expected number of vehicles reaching any given urban location. We use this information to infer the better locations for deploying the roadside units in order to maximize the number of trips experiencing at least one vehicle-to-infrastructure contact opportunity. Our deployment strategy is evaluated using three distinct scenarios with growing complexity: (a) theoretical road network; (b) real road network and synthetically generated traffic; (c) real road network and realistic traffic.

Since traffic fluctuates, an architecture based only on stationary roadside units is unable to properly support the network operation all the time. Thus, we also investigate the benefits of dynamic infrastructure deployment strategies. Although traffic fluctuates over time, such fluctuation is somehow limited by the underlying (and almost) fixed road network. Thus, we propose a hybrid architecture based on stationary and mobile roadside units. Our investigation demonstrates that approximately 60% of the roadside units may be stationary. In order to address traffic fluctuations, the mobile roadside units must travel at an average speed of 5.2km/h, which demonstrates the feasibility of incorporating the mobile infrastructure into public transportation vehicles.

List of Figures

List of Tables

List of Algorithms

Table 1. List of Symbols.

Symbol	Meaning
$M_{i,j}$	Bi-dimensional matrix holding number of vehicles at intersection of roads i and j;
$P_{i,j}$	Stochastic matrix holding turning ratio. $P_{i,j}$ indicates the probability that vehicles traveling on road i turn into road j;
$P^{u}_{i,j}$	Bi-dimensional matrix holding the likelyhood of vehicles moving **up** at intersection (i, j);
$P^{d}_{i,j}$	Bi-dimensional matrix defined over $1..n, 1..n$ holding the likelyhood of vehicles moving **down** at intersection (i, j);
$P^{r}_{i,j}$	Bi-dimensional matrix defined over $1..n, 1..n$ holding the likelyhood of vehicles moving **right** at intersection (i, j);
$P^{l}_{i,j}$	Bi-dimensional matrix defined over $1..n, 1..n$ holding the likelyhood of vehicles moving **left** at intersection (i, j);
$C^{P,k}_{M,\Gamma}$	Represents the projection of the flow for all roads belonging to the road network considering the deployment of the k^{th} roadside unit belonging to the solution set Γ. $PF^{P}_{m,\gamma}$ $\forall m \in M$, where γ is the k^{th} element of Γ;
Λ	Bi-dimensional matrix holding the projected flow of vehicles;
α	Number of roadside units to be deployed;
Γ	Solution set, i.e., intersections of urban cells receiving roadside units;
S	Bi-dimensional matrix holding the sorted values of M. $\|S\| = \|M\|$;
ϱ	Geographical map of the road network;
ψ_x	Number of horizontal urban cells of the road network;
ψ_y	Number of vertical urban cells;
Δ	Grid representing road network;
T	Mobility trace;
τ	Element of T;
G	Mobility trace converted to grid coordinates;
K	Set of vehicles (Chapter 6);
C	Road network partitions (Chapter 6);
M	Trajetories (Chapter 6);
A	Urban cells receiving roadside units (Chapter 6);
V	'Covered' vehicles;
λ	Threshold of low traffic; (Chapter 7)
μ	Interval (in minutes) for detecting low traffic (Chapter 7);
η	Maximum size of the vehicle queue (Chapter 7);
ϑ	Traffic light Id (Chapter 7);

Contents

Chapter 1

Introduction

Nowadays, traffic is one of the most important urban issues, and it is becoming unmanageable in large metropolitan areas. As an example, the amount of vehicles increases annually by rates of at least 10% in most developing countries, thus doubling the number of cars every seven years (Gakenheimer, 1999). Furthermore, traffic raises other areas of concern, such as environmental pollution, public health and safety. An Intelligent Transportation System would help to alleviate the problems of the transportation sector. By embedding computing devices into cars, roads, streets, and transit equipment such as street signs, radars, traffic cameras and others, we will be capable of 'digitalizing' the transportation system (Macedo et al., 2012; Oliveira et al., 2013). Applications are endless: driver assistance for faster, less congested and safer roads; more efficient use of the transportation system; more efficient planning of routes and control of the traffic flow; secure and greener traffic through digital driver assistance; and better planning and evolution of the system as a whole due to the availability of historical data, based on traffic and utilization trends detected via data mining techniques. As indicated by the European Transportation Policy (2001), the use of Intelligent Transportation Systems is one of the key technologies for improving the safety, efficiency and environmental friendliness of the transport industry.

An Intelligent Transportation System is grounded on a sophisticated communication network receiving data from several entities composing the traffic system. Data is processed and translated into useful information and recommendations to assist users of the transportation system and transit authorities. Such sophisticated communication network is commonly referred as a vehicular network (Hartenstein and Laberteaux, 2008; Yousefi et al., 2006; Jakubiak and Koucheryavy, 2008).

A **vehicular network** is a network connecting vehicles in order to provide a platform for future deployment of large-scale and highly mobile applications. Mobility

is probably the most intrinsic and distinguished feature of vehicular networks. In such dynamic network, localizing vehicles is a necessary challenge (Boukerche et al., 2008) in order to maximize the network benefits. Future position of vehicles is almost non-deterministic and previous warning of trajectories by vehicles is an unrealistic assumption. Given the automobile's role as a critical component in peoples' lives, embedding software-based intelligence into them has the potential to drastically improve drivers' quality of life (Gerla et al., 2006). Leveraging wireless communication in vehicles has fascinated researchers since the 1980s (Kawashima, 1990). A few years later, as a consequence of the cellular revolution, voice services have become commonplace and ubiquitous, and the attention of researchers has shifted to wireless communication (Frenkiel et al., 2000).

In a vehicular network, communication may happen in a direct and completely distributed basis where vehicles exchange messages without any support infrastructure (vehicle-to-vehicle communication). Such kind of ad hoc communication has been extensively studied by the research community (Hartenstein and Laberteaux, 2008; Jakubiak and Koucheryavy, 2008). However, the ad hoc communication may become inefficient due to the high mobility of vehicles, leading to very short contact times possibly reducing the network throughput. Additionally, communication also suffers in sparse areas such as highways, rural zones and low peak hours in city due to the lack of communicating pairs. High mobility also makes routing far complicated as we lack reliable means to infer future position of vehicles. Although communication may take place in a vehicle-to-vehicle basis, **research demonstrates that a minimum support infrastructure may largely improve the overall efficiency of the vehicular network** (Kozat and Tassiulas, 2003; Gerla et al., 2006; Banerjee et al., 2008; Reis et al., 2011; Mershad et al., 2012; Wu et al., 2012b).

1.1 Infrastructure for Vehicular Networks

In the context of vehicular networks, **infrastructure is a set of specialized communication devices supporting the network operation**. Common properties include (but are not restricted to) network centrality, communication bandwidth, storage space and high availability. Because these devices were initially conceived to be located at roadsides, they are commonly referred to roadside units (RSUs). Roadside units may provide several functionalities such as: (i) Broadcast functionalities (Viriyasitavat et al., 2010); channel allocation (Harigovindan et al., 2014); caching (Liang and Zhuang, 2012); content download (Malandrino et al., 2011; Liu et al.,

2013); data dissemination strategies (Lochert et al., 2007); data aggregation (Bruno and Nurchis, 2013); data scheduling (Shumao et al., 2009); gaming and streaming (Palazzi et al., 2008, 2009); gateway functionalities (Wu et al., 2005a; Gerla et al., 2006; Liang and Zhuang, 2012); hand-off optimizations (Ramani and Savage, 2005; Brik et al., 2005; Amir et al., 2006); localization of vehicles (Boukerche et al., 2008; Rosi et al., 2008); quality of service assurance (Luan et al., 2012); real time support (Korkmaz et al., 2010; Liu and Lee, 2010; Abdrabou et al., 2010); routing (Borsetti and Gozalvez, 2010; Annese et al., 2011; Mershad et al., 2012); security, privacy and reputation (Marmol and Perez, 2012; Fernandez Ruiz et al., 2010; Plobl and Federrath, 2008; Souza et al., 2009); and, support to multi-hop communication (Zhang et al., 2012).

Several technologies and devices may serve as roadside units. Banerjee et al. (2008) present an in-depth discussion and comparison of such technologies. Although most of the works consider stationary infrastructure, several researches (Jerbi et al., 2008; Luo et al., 2010; Annese et al., 2011; Mishra et al., 2011; Tonguz and Viriyasitavat, 2013; Sommer et al., 2013) propose mobile architectures (public transportation buses, cabs, ordinary vehicles). There are also proposals considering the use of low cost devices as an infrastructure (Luan et al., 2013), while another proposals consider the use external devices serving the vehicular network such as public available Wi-Fi connections (Marfia et al., 2007).

Throughout this thesis we conducted a Systematic Literature Review (SLR) on infrastructure-based vehicular networks from Jan. 2003 up to Feb. 2014 where we mapped the research evolution over the years, and the state-of-the-art in:

(a) Infrastructure deployment in vehicular networks;

(b) Vehicle-to-infrastructure communication;

(c) Architectures of infrastructure-based vehicular networks.

Our Systematic Literature Review captured 30 works addressing aspects of infrastructure deployment. When we analyze all these efforts we conclude that they usually consider unrealistic road networks or unrealistic mobility, and such lack of realism implies in severe impreciseness in results reported by these works[1] (Fiore and Härri, 2008).

[1]Fiore and Härri (2008) present an in-depth analysis of the topological properties of a vehicular network and found that simulation results are strongly affected by the mobility model, and they question the validity of studies conducted under unrealistic car mobility scenarios.

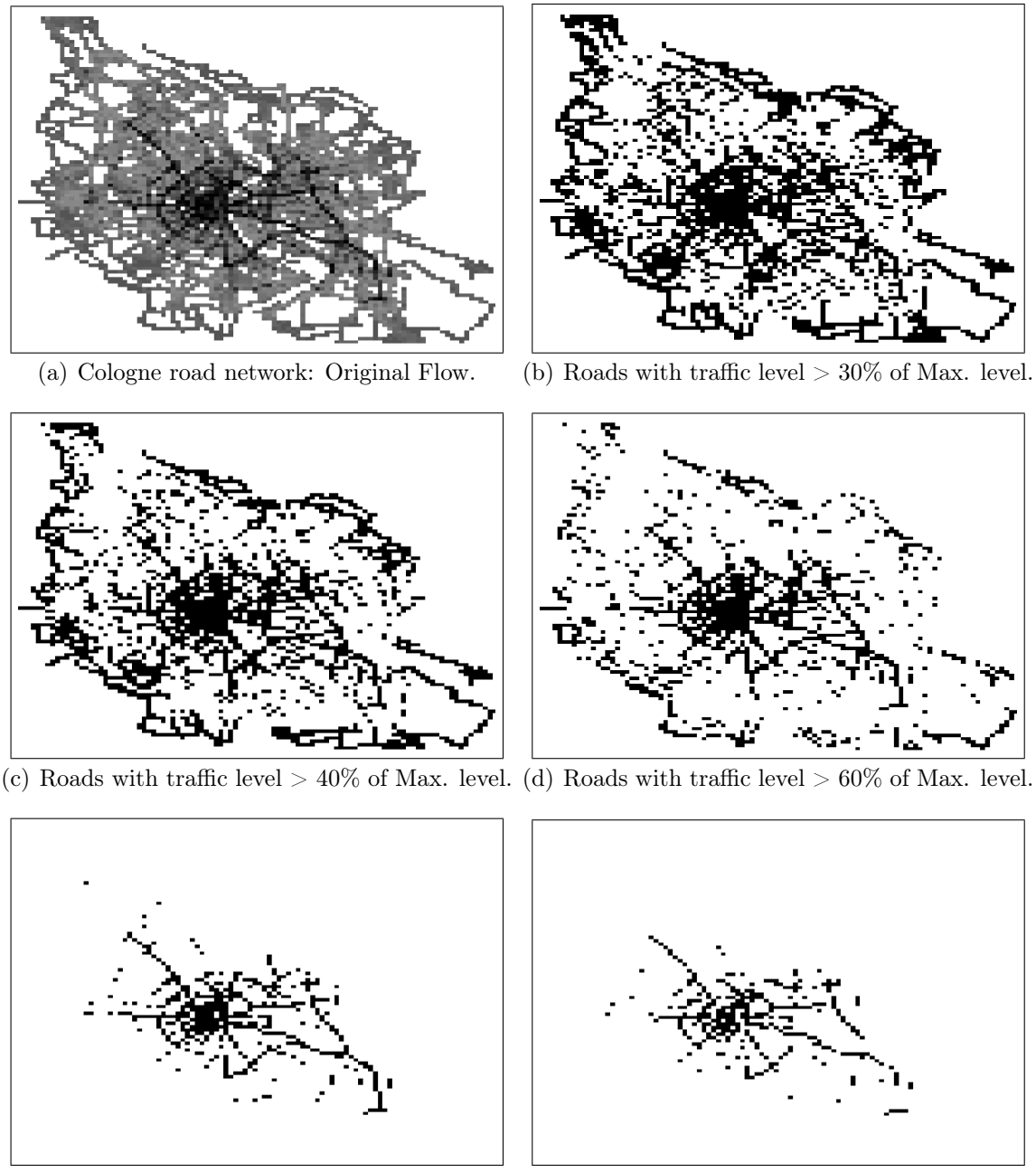
1.2 Infrastructure Deployment

Deployment of infrastructure is one of the most critical decisions when designing a vehicular network. **Infrastructure deployment** is the task of defining the exact position of each roadside unit within the road network. A misleading deployment means waste of valuable resources and degradation of the network performance.

Infrastructure deployment for vehicular networks is an open problem. Given that high mobility is, perhaps, the most distinguished feature of vehicular networks, it must be the start point of a deployment strategy. Typically, deployment algorithms propose alternative ways to identify the most populated locations within the road network. Those selected locations receive roadside units. Although placing roadside units in densest places may seem reasonable at a first glance, the assumption fails when we consider that those vehicles composing the dense regions are originated from nearby, and the dense region is created as a result of merging flows. Figure 1.1 presents the realistic flow of vehicles in the city of Cologne (Germany), and we notice: (a) flows creating dense areas; and, (b) connectivity between dense areas. Intensity of black color indicates traffic level. From Figure 1.1(a) to Figure 1.1(f), we present the remaining road network after eliminating roads below an increasing minimum traffic threshold.

Selecting the densest places is a correct decision when nodes composing the network are stationary (or moving at very low speeds). But when dealing with high speed mobile nodes (i.e., vehicles) we must consider the mobility and the flow characteristics. Typically, dense regions do not appear as isolated islands, but they result from merging flows converging to attraction areas. Such issue indicates that dense regions tend to appear somehow interconnected, and vehicles traveling nearby dense areas have high probability of joining the main flow. Thus, when we consider just the concentration of vehicles to position the roadside units, we may incur in redundant coverage by deploying roadside units covering the same flow twice or more. Thus, we investigate the benefits of incorporating mobility information into deployment algorithms.

Deployment proposals typically assume two levels of mobility to solve the allocation of roadside units: (i) full mobility information; (ii) no mobility information at all. Works assuming previous knowledge of vehicles trajectories employ a full mobility information model. However, full mobility is not a feasible assumption in real deployments. On the other hand, works proposing alternative strategies to identify locations accounting for higher volume of vehicles do not assume any mobility information at all. In this work we propose a third level of mobility information: we use vehicles' migration ratios between adjacent urban cells.



(a) Cologne road network: Original Flow.
(b) Roads with traffic level > 30% of Max. level.
(c) Roads with traffic level > 40% of Max. level.
(d) Roads with traffic level > 60% of Max. level.
(e) Roads with traffic level > 80% of Max. level.
(f) Roads with traffic level > 90% of Max. level.

Figure 1.1. Figure illustrates merging flows converging to attraction areas.

Formally, we can formulate the deployment of infrastructure aiming to maximize the number of distinct vehicles contacting the infrastructure as follows:

> Let $G = (V, E)$ be a graph where V represents the set of roads of a given road network and E represents roads intersections. Let $A = \{a_1, a_2, ..., a_h\}$ represent the set of h vehicles. Available number of roadside units is given

by α. $\Gamma \leftarrow \emptyset$ is our solution set, i.e., the position of each deployed roadside unit. $C \leftarrow \emptyset$ represents the set of 'covered' vehicles, i.e., vehicles that have already reached any roadside unit during the trip. $\forall a \in A$ *do* $C \leftarrow C \cup a$ if a has crossed at least one of locations composing Γ: vehicle is considered 'covered' if it reaches any roadside unit. Roadside units are deployed at locations Γ.

Our goal is to find $\Gamma \subseteq G$ where $|\Gamma| \leq \alpha$ and $|C|$ is maximum.

Our Systematic Literature Review (Chapter 2) also reveals that previous deployment works do not agree on a strategy to represent the road network, and just two works (Cheng et al., 2013; Patil and Gokhale, 2013) consider basic concepts of Geographical Information Systems (GIS). Lacking a standard representation of road networks makes comparison of results far complicated. Thus, we propose a strategy to represent road networks of arbitrary complexity.

We also exploit the Probabilistic Maximum Coverage Problem (PMCP) for infrastructure deployment over vehicular networks. A deployment strategy that selects the densest locations to receive the roadside units is, in fact, proposing a solution for Maximum Coverage Problem. **Maximum Coverage Problem** considers a collection of sets defined over a domain of elements. These elements are distributed over a given number of sets. Each element may appear in more than one set. Sets of elements are static, thus elements do not migrate between sets. Goal is to select those α sets presenting maximum cardinality. Given that elements do not migrate between sets, Maximum Coverage Problem is not suitable to handle mobility. On the other hand:

Probabilistic Maximum Coverage Problem considers a collection of sets defined over a set of elements. These elements have a probability of belonging to a given set. Sets of elements are not static because elements are not tied to sets. Goal is to select those α sets presenting maximum **expected** cardinality. We highlight the application of the Probabilistic Maximum Coverage Problem as a contribution of this work because, as far as we know, Probabilistic Maximum Coverage Problem is exploited in just one work in the literature: Fan and Li (2011) apply Probabilistic Maximum Coverage Problem in order to maximize information propagation in social networks. Ultimately, information propagation is the 'mobility of information'.

1.3 Objectives and Intended Contributions

We propose a novel deployment algorithm that uses partial mobility information in order to position the roadside units within a road network of arbitrary topology. Partial

mobility information means that we do not rely on individual vehicles trajectories. Only information we have are migration ratios between adjacent urban cells. Our solution models the allocation of roadside units as a Probabilistic Maximum Coverage Problem, a probabilistic instance of the traditional Maximum Coverage Problem, and we consider the position of each vehicle to be no longer deterministic, but a probabilistic position given by function $f : \{c_1, c_2, P\} \longrightarrow \mathbb{R}$ that returns the probability that a vehicle located at urban cell c_1 will migrate to urban cell c_2 considering the mobility model P. Urban cells are micro-areas composing the road network. Union of all urban cells is the road network itself. Chapter 4 presents an in-depth discussion of urban cells, road network partitioning, and experimental results considering the road networks partitioning.

Splitting the road network into a set of discrete urban cells simplifies the detection of sources, destinations, and migration ratios. Basic mobility patterns are highlighted by splitting the road network into just a few urban cells. More sophisticated mobility patterns are highlighted by increasing the number of urban cells covering the road network. Splitting the road network into urban cells is the core of our proposed strategy to represent road networks of arbitrary topology.

Thus, our intended contribution with this thesis is:

Design of a deployment algorithm employing a new paradigm of mobility information.

Next section presents an overview of our solution.

1.4 Overview of Our Solution

In this work[2] we design a novel infrastructure deployment algorithm for vehicular networks. Our algorithm uses a mobility model based on global behavior, i.e., we exploit the vehicles' migration ratios between urban cells composing the road network. Our work is motivated by the study of Trullols et al. (2010), which demonstrates that when we know the trajectories of vehicles in advance, we achieve close-to-optimum deployment performance. However, when we lack information of vehicles trajectories, performance of Maximum Coverage Problem is strongly affected. Because full knowledge of vehicles trajectories implies in several privacy (and also practical) issues, we have turned our attention to improve the performance of Maximum

[2]Data and Source Code available at: `https://dl.dropboxusercontent.com/u/15930967/sim-cms.rar`

Coverage deployment by using a mobility model describing the global behavior, instead of individual behavior.

In **Chapter 2** we present a Systematic Literature Review of works published from Jan. 2003 to Feb. 2014 in IEEExplore, ACM Digital Library and Elsevier Science Direct in order to characterize the evolution of the research in infrastructure-based vehicular networks.

In **Chapter 3** we develop our deployment algorithm in a theoretical grid-based road network with randomized flows (Chun et al., 2010; Larson, 1982; McNeil, 1968; Gerlough and Schuhl, 1955). We assume roadside units always positioned at roads intersections. We model the position of each vehicle as a function $f : \{c_1, c_2, P\} \longrightarrow \mathbb{R}$ that returns the probability that a vehicle located at urban cell c_1 migrates to urban cell c_2 considering the mobility model P. Our mobility model lies on the assumption that vehicles do not have a deterministic position. Instead, vehicles have a probability of being at a given position in a given instant of time. By using the mobility model we define the deployment of infrastructure as a Probabilistic Maximum Coverage Problem. However, because we do not assume previous knowledge of vehicles trajectories, we are not able to truly solve neither PMCP, nor MCP. Instead of it, we propose constructive heuristics to approximate both problems solutions.

In **Chapter 4** we generalize our deployment algorithm to handle real road networks, and we also perform a worst-case analysis of the deployment performance considering a PMCP-based approach, and a MCP-based one. We use the road network of a Brazilian city[3] using synthetically generated flows in order to perform a worst-case analysis of deployment efficiency. We partition the road network into a set of adjacent same size urban cells. Once the road network is partitioned we discard the real road network and we manipulate the flow between adjacent urban cells. We also evaluate the benefits of using **dynamic deployment** of roadside units. Although dynamic deployment is not possible with stationary infrastructure, we have already mention the existence of several works (Jerbi et al., 2008; Luo et al., 2010; Annese et al., 2011; Mishra et al., 2011; Tonguz and Viriyasitavat, 2013; Sommer et al., 2013) proposing the adoption of a virtual-and-mobile infrastructure to support the vehicular communication. Thus, contributions of this chapter are: (a) characterizing the efficiency of partitioning the road network; (b) characterizing the worst-case deployment performance of a PMCP-based and MCP-based approaches.

In **Chapter 5** we evaluate our deployment algorithm using a real road network (750km^2) and a realistic vehicular trace. The results demonstrate 89.6% of the vehicles

[3]Ouro Branco, Minas Gerais, Brazil.

experience at least one V2I contact by covering just 2.5% of the entire road network. Now we compare PMCP-based to MCP-based and also MCP-g (greedy solution for MCP).

After that, we reevaluate the partition technique considering MCP-g presented in Algorithm 1. MCP-g is the greedy solution for the Maximum Coverage Problem. Thus, it relies on previous knowledge of vehicles trajectories. Our selection of MCP-g is based on the fact that MCP-g achieves close-to-optimum coverage (Section 5.5). We propose the deployment of roadside units covering just 0.5% of the road network. Thus, after partitioning the road network we select the 0.5% of the most promising urban cells to receive the roadside units.

Then, we propose an Integer Linear Programming Formulation (Opt_{dv}) for the deployment. We compare our PMCP-based approach, MCP-based, and Opt_{dv} through several experiments. Our experiments reveal that our Integer Linear Programming Formulation is able to solve instances presenting large number of roadside units. Thus, we have assumed the deployment of α=500 roadside units (coverage of 5% of the entire road network). We were not able to solve instances presenting lower number of roadside units, namely α=250, and α=50 without offering initial solutions to the model. We have evaluated our Integer Linear Programming Formulation offering initial solutions computed using the well-know greedy solution for MCP (MCP-g) presented in Algorithm 1. CPLEX was not able to improve the MCP-g solution, but it outputs that optimal solution is at most 3.5% above the MCP-g solution.

In **Chapter 6** we investigate an architecture composed of stationary and mobile roadside units. It is a common sense that traffic fluctuates: thus, an architecture employing just stationary roadside units might not be able to properly support the network operation all the time. Similarly, an architecture composed just of mobile roadside units makes a few sense when we consider that the road network does not change that often. In other words, traffic fluctuations are limited by its subjacent road network. As major roads counts on higher transportation capacity, they tend to be very popular routes, and they are natural candidates for receiving stationary roadside units. And mobile roadside are highlighted as ideal solutions to handle in-borders traffic. In order to improve the deployment performance, we may rely on hybrid deployment strategies employing sets of stationary and mobile roadside units. Stationary roadside units act as a main backbone for data dissemination covering the most important regions of an urban area (i.e., regions known as always presenting relevant traffic). On the other hand, we may rely on mobile roadside units to address traffic fluctuations.

We also present three appendices: In Appendix A we exploit an Intelligent Transportation Systems Applications: LaNPro is a module of a smart traffic light

designed to operate under low traffic conditions usually found in small cities and during early hours of morning. In Appendix B we present our publications and awards. Finally, in Appendix C we expand our systematic literature review.

1.5 Overview of the Next Chapter

In the next chapter we present our Systematic Literature Review of infrastructure-based vehicular networks.

Chapter 2

Background

In this chapter we review and characterize works published in IEEE, ACM and Science Direct (Elsevier) from Jan. 2003 to Feb. 2014 addressing **infrastructure-based vehicular networks**. We have selected a set of 68 articles published in conferences and journals, and our goal is to identify the solutions proposed in the literature through a Systematic Literature Review (SLR). This chapter is organized as follows: Section 2.1 presents our methodology. Sections 2.2 to 2.4 describes the selected works. Section 2.6 concludes the chapter.

2.1 Methodology

Studies of this research have been collected in electronic databases meeting the following criteria: (a) peer reviewed articles; (b) search engine by keywords per field; (c) full access to articles; and (d) recognized reputation in publishing high quality content. Resulting list of selected databases is: ACM Digital Library (`http://portal.acm.org/`); IEEEXplore (`http://www.ieeexplore.ieee.org/`); Elsevier Science Direct (`http://www.sciencedirect.com/`). We have defined our period of interest as Jan. 2003 to Feb. 2014, and the search string as:

> **((infrastructure OR roadside OR rsu OR dissemination points OR access points) AND (vehicular networks OR vanet))**

Using the search string we have retrieved all articles with matches in title, abstract or keywords using the search engine of each publisher (ACM, IEEE, Elsevier). In order to validate the search string we have conducted an investigative review in order to identify false-negatives (articles related to our target but not retrieved by the search

string). The investigative review is performed in three steps: (i) Search via Google Scholar[1]; (ii) Search via Mendeley[2] Tool; (iii) Manual inspection of the bibliography of the selected articles from 2011 up to 2014.

By following the steps of a SLR (Kitchenham, 2004; Kitchenham and Charters, 2007), we have established a four-step process with different review processes, namely: (a) remove duplicate papers; (b) remove papers not dealing with infrastructure aspects of vehicular networks; (c) eliminate papers by scanning through the full text in order to check whether the inclusion or exclusion criteria were met; (d) full text analysis in articles meeting inclusion criteria. Inclusion and exclusion criteria shown below were used to narrow the search to relevant papers.

- **Inclusion criteria:** papers describing researches on infrastructure-based vehicular networks;
- **Exclusion criteria:** short papers, editorials, posters, introductions of keynotes, mini-tracks, studies in languages other than English.

In the following sections we describe the selected articles: Section 2.2 presents **deployment** works. Section 2.3 presents **architecture** works. Section 2.4 presents **communication** articles.

2.2 Deployment

In this section we present works addressing the deployment of infrastructure for vehicular networks. Until 2006, the research community is focused on low level aspects of ad hoc communication. Initial deployment efforts focus on bringing Internet into vehicles. Soon the research community realizes that a communication infrastructure is able to offer much more than Internet access. In a attempt to reduce the high costs of the deployment, the research community evaluates: (i) use of public available Wi-Fi networks (spread over urban centers) to support the vehicular communication; (ii) design of virtual-and-mobile infrastructures composed of vehicles and buses; (iii) the use of cheap devices acting as roadside units.

In 2007-2008 we notice the initial efforts presenting deployment algorithms. Such efforts propose strategies maximizing the vehicle-to-infrastructure contact probabilities (Li et al., 2007), or content distribution (Rosi et al., 2008). There are also analytic

[1] `http://scholar.google.com`

[2] Mendeley provides a combination of a desktop and a website for sharing scientific articles among team members. `http://www.mendeley.com`

studies (Nekoui et al., 2008), and the application of genetic programming (Lochert et al., 2008). Figure 2.1 presents the number of published articles per year. The x-axis indicates year, while y-axis indicates number of published articles. Figure 2.2 presents the strategy used to solve the deployment.

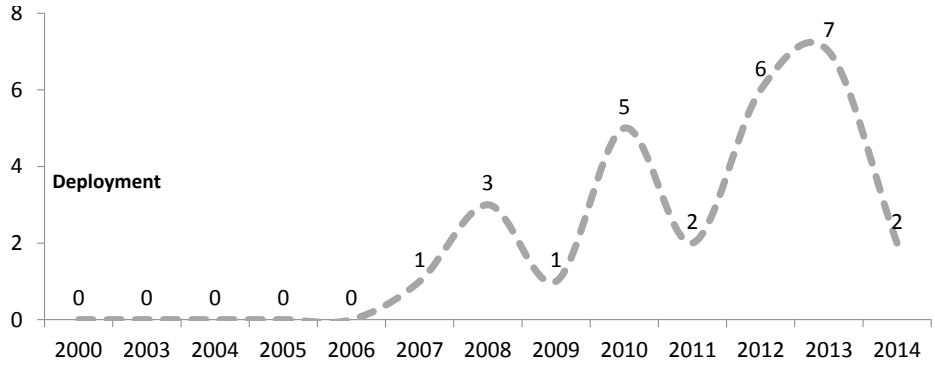


Figure 2.1. Number of articles addressing 'deployment' of infrastructure for vehicular networks: The x-axis indicates year, while y-axis indicates number of published articles.

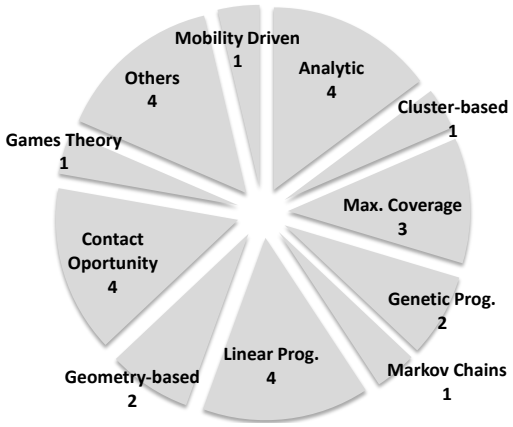


Figure 2.2. Organization of works addressing 'deployment' in infrastructure-based vehicular networks.

- **Linear Programming**: efforts proposing a linear programming (Gass, 1958; Dantzig, 1998; Luenberger, 1973; Simonnard and Jewell, 1966) formulation to solve the allocation of the infrastructure. Linear programming is a method to achieve the best outcome in a mathematical model whose requirements are represented by linear relationships. Linear programming is a special case of mathematical programming. More formally, linear programming is a technique for the optimization of a linear objective function, subject to linear equality and

linear inequality constraints. Its feasible region is a convex polyhedron, which is a set defined as the intersection of finitely many half spaces, each of which is defined by a linear inequality. Its objective function is a real-valued affine function defined on this polyhedron. A linear programming algorithm finds a point in the polyhedron where this function has the smallest (or largest) value if such a point exists;

- **Maximum Coverage**: efforts modeling the allocation of infrastructure as a Maximum Coverage Problem. Maximum Coverage is a classical problem in Computer Science. Basically, we have a collection of sets defined over a domain of elements. These elements belong to some sets. One element may belong to more than one set. The goal is to select a collection of sets maximizing the cardinality of elements. A detailed discussion of the problem is found in (Cormen et al., 2001);

- **Analytic Studies**: efforts proposing analytic formulations to model aspects of the deployment, typically delay-bounded strategies for vehicle-to-vehicle and vehicle-to-infrastructure communications;

- **Cluster-based**: efforts proposing deployment strategies based on data clustering (Zaki and Meira Jr, 2014; Kaufman and Rousseeuw, 2009). Clustering is the task of grouping a set of objects in such a way that objects in the same group are more similar to each other than to those in other groups;

- **Contact Opportunities**: efforts proposing deployment strategies in order to maximize the vehicle-to-infrastructure contact probability;

- **Markov Chains**: efforts proposing Markov Chains (Chung, 1967; Markov, 1971; Meyn and Tweedie, 2009; Kemeny and Snell, 1960) to model the encounters of vehicles and the infrastructure. A Markov chain is a mathematical system that undergoes transitions from one state to another on a state space. It is a random process usually characterized as memoryless: the next state depends only on the current state and not on the sequence of events that preceded it;

- **Game Theory**: efforts using concepts of Game Theory (Harsanyi and Selten, 1988; Gibbons, 1992; Osborne and Rubinstein, 1994; Myerson, 2013) to model some aspect of the deployment. Game theory is a study of strategic decision making. Specifically, it is the study of mathematical models of conflict and cooperation between intelligent rational decision-makers (Myerson, 2013). Game theory is mainly used in economics, political science, and psychology, as well as

logic, computer science, and biology. The subject first addressed zero-sum games, such that one person's gains exactly equal net losses of the other participant or participants. Today, however, game theory applies to a wide range of behavioral relations, and has developed into an umbrella term for the logical side of decision science;

- **Mobility-Driven**: efforts proposing the use of vehicular mobility to assist the deployment activity;
- **Genetic Programming**: efforts using genetic programming (Koza, 1992; Banzhaf et al., 1998) in order to find out better solutions (considering a given objective function). Genetic programming is an evolutionary algorithm-based methodology inspired by biological evolution to find computer programs that perform a user-defined task. Main operators used in evolutionary algorithms are crossover and mutation;
- **Geometry-based**: efforts applying geometry concepts in order to deploy the infrastructure.
- Others.

Sections 2.2.1 to 2.2.11 detail these works.

2.2.1 Analytic Studies

Nekoui et al. (Nekoui et al., 2008) propose an infrastructure for vehicular networks based on the conventional definition of the transport capacity. Authors develop a mathematical model where the destination nodes are chosen at random by the source nodes. Authors study the effect of infrastructure node deployment in the capacity of vehicular networks, and using analytical expressions they show that exploiting any number of infrastructure nodes beyond a certain amount enhances the achievable capacity. Although the authors propose to handle arbitrary topologies, they assume several simplifications in the mobility model.

Alpha Coverage (Zheng et al., 2009) provides worst-case guarantees on the interconnection gap while using significantly fewer roadside units. A deployment of roadside units is considered α-covered if any simple path of length α on the road network meets at least one roadside unit. Authors compare the α-coverage with random deployment of roadside units.

Sou and Tonguz (Sou and Tonguz, 2011) investigate the allocation of roadside units along a highway. They analyze the performance of the roadside units by taking

into account important issues like vehicle deceleration, channel congestion, different beacon frequencies, hidden node problem, and multi-lane traffic. The results indicate that on a 300km highway the rehealing delay is reduced by 70%, whereas the average number of rehealing hops is reduced by 68.4% when deploying 50 roadside units compared to an operation with no roadside unit. Authors conclude that: (i) the deployment of a small number of roadside units may achieve a substantial improvement when the vehicular network is sparse; (ii) the roadside units prevent the occurrence of the broadcast storm; and (iii) roadside units improve the message penetration time and packet delivery ratio, while reducing the delay.

Sou (Sou, 2010) also addresses the placement of roadside units at rural areas and roadways where the solutions must deal with the low density of vehicles and very large areas to be covered. The author proposes the deployment of the roadside units equally distanced from each other along a roadway enabling some roadside units to enter the power-saving mode in order to optimize the energy consumption. The work presents an algorithm to select the roadside units to enter the power-saving mode.

2.2.2 Clustering Strategies

Kchiche and Kamoun (Kchiche and Kamoun, 2010) apply clustering techniques to solve the roadside units' allocation through a greedy algorithm based on the centrality of group in order to select the better locations for the infrastructure. The algorithm aims to maximize the performance of the message distribution system by reducing the global delay and the messages communication overhead. The authors also demonstrate that the centrality and the equidistance of the infrastructure are important features to improve the quality of the coverage. When considering just the ad-hoc communication, authors report an end-to-end delay of several minutes.

2.2.3 Contact Opportunity

Li et al. (Li et al., 2007) propose a placement strategy for a set of access points to access a single gateway in an open space, similar to the base station placement in cellular systems. The goal is to minimize the power consumption and the average number of hops from access points to gateways under the assumption of full coverage. Gateways connect access points to the Internet, whereas minimizing the average number of hops from access points to gateways. Every vehicle is considered connected to an access point. Vehicle speed, density or movement patterns have not been considered.

Zheng et al. (Zheng et al., 2010) present the evaluation of a deployment strategy considering the contact opportunity. Contact opportunity measures the fraction of distance or time that a vehicle is in contact with the infrastructure. Authors argue that such metric is closely related to the quality of data service that a mobile user might experience while driving. Authors also propose a deployment algorithm intended to maximize the worst case contact opportunity under budget constraints. The solution is evaluated using computer simulations and a testbed in a university campus. Authors compare the results with two baseline algorithms: (i) **Uniform Random Sampling** selects the location of the access points at random; (ii) **Max-Min Distance Sampling** (Teng, 1995) starts at a randomly selected location, and at each step allocates a new access point in order to maximize the minimum graph distance (in terms of shortest paths) from the elements already selected. Experimental results show that the deployment strategy achieves more than 200% higher minimum contact opportunity, 30%-100% higher average contact opportunity, and a significantly improved distribution of average throughput when compared to Uniform Random Sampling and Max-Min Distance Sampling.

Lee and Kim (Lee and Kim, 2010) propose a greedy heuristic to place the infrastructure aiming to improve the vehicles connectivity while reducing the disconnections. The heuristic counts the amount of reached vehicles at each intersection by considering the transmission range of the roadside units. Each intersection is considered as a potential roadside unit location. Optimal locations are selected based on the number of vehicle reports (per minute locations reported by taxis to telematics system) received within the communication range of each roadside unit. Placement scheme considers only the taxi location reports, and it does not take into account speed, nor density of vehicles.

Chi et al. (Chi et al., 2013) propose three optimal algorithms to allocate the roadside units: Greedy, Dynamic and Hybrid algorithms. Authors assume: (i) placing roadside units preferentially at important intersections; (ii) allocating roadside units until every intersection is covered; (iii) distributing the roadside units as even as possible. The relevance of each intersection is evaluated using traffic factors, including vehicles density, intersection popularity, and intersection particularity. The greedy algorithm simply deploys the roadside units at intersections in a descending order of the intersection priority. Dynamic algorithm, however, concentrates on the even distribution of roadside units. The hybrid algorithm combines the greedy and the dynamic methods.

2.2.4 Markov Chains

Liu et al. (Liu et al., 2013) propose a new roadside units' deployment strategy for file downloading in vehicular networks. The encounters between vehicles and roadside units are modeled as a time continuous homogeneous Markov chain. The road network is modeled as a weighted undirected graph, and the authors propose an algorithm for the deployment of roadside units.

2.2.5 Game Theory

Filippini et al. (Filippini et al., 2012) apply games theory to characterize the better strategy for competing providers to deploy the roadside units in order to maximize the revenue. The authors derive preliminary results evaluated via simulations. They consider a scenario with two operators deploying roadside units for distributing content along a road of length D. Each roadside unit is characterized by a coverage range R, which defines its service area, and by an application level goodput c for content delivery. The goodput measure depends on the wireless technology, and on the communication protocols used for content delivery.

2.2.6 Genetic Algorithm

Lochert et al. (Lochert et al., 2008) study how the infrastructure may be used to improve the travel time of data over large distances. The authors present a multilayer aggregation scheme defining landmarks. Cars passing landmarks record time travel, which is aggregated to infer the time travel between more distant landmarks. These aggregation steps are performed by the cars themselves in a completely decentralized basis whenever information that is a suitable basis for forming an aggregate becomes locally available. Minimal initial deployment of roadside units is handled by a genetic algorithm based on the travel time savings. Cavalcante et al. (Cavalcante et al., 2012) apply genetic programming to solve the deployment of dissemination points in vehicular networks. They start with an initial set of possible solutions that are combined across generations until some stop condition is reached. The authors model the problem as a Maximum Coverage and they impose a time limit.

2.2.7 Geometry-based Heuristics

Cheng et al. (Cheng et al., 2013) propose a geometry-based coverage strategy to handle the deployment problem over urban scenarios. By taking the shape and area

of road segments into account, the scheme suits different kinds of road topologies and effectively solves the maximum coverage problem. Patil and Gokhale (Patil and Gokhale, 2013) propose a Voronoi (Aurenhammer, 1991) diagram-based algorithm for the effective placement of roadside units using the packet delay and the loss as criteria. Authors seek to provide a collaborative mechanism for dynamic resources management in vehicular networks, and to provide quality of service assurances to applications. The collaboration between vehicles and roadside units is enabled through a vehicle-to-infrastructure network. They have collected population census and use this population information within SUMO's ActivGen API to generate traffic data mimicking the real-world. Results are compared to **RSUs Evenly Spread** and **RSUs at most Busiest Signals**. Voronoi diagram-based deployment algorithm results in less packet delay/packet loss.

2.2.8 Linear Programming

Aslam et al. (Aslam et al., 2012) use Binary Integer Programming to solve the allocation of infrastructure: they eliminate minor roads and model major roads as a grid. Authors present two different optimization methods for placing a limited number of roadside units into an urban region: (i) analytical Binary Integer Programming (BIP); (ii) novel Balloon Expansion Heuristic (BEH). BIP method utilizes branch and bound approach to find an optimal analytical solution whereas BEH method uses balloon expansion analogy to find an optimal or near optimal solution. Authors conclude that the BEH method is more versatile and performs better than BIP method in terms of computational cost and scalability.

Wu et al. (Wu et al., 2012a) focus on a highway scenario with multiple lanes, exits and intersections along the road. Vehicles may communicate with the infrastructure or use multi-hop relay when out of the infrastructure's transmission range. The authors model the deployment of roadside units as a Integer Linear Programming considering both strategies of communication such that the aggregate throughput in the network is maximized. Authors also model the impact of the wireless interference, vehicle population distribution, and vehicle speeds. The model is evaluated via ns-2 and VanetMobiSim[3] simulations. The multi-hop relay allows vehicles to deliver packets to the infrastructure (ahead or backward), according to the smallest hop count. The results demonstrate that scheme overcomes uniformly distributed placement.

Liang et al. (Liang et al., 2012) study the deployment of infrastructure in 2-D by formulating an optimization problem and solving it using an Integer Linear

[3]VanetMobiSim. `http://vanet.eurecom.fr/`

Programming Formulation. The proposed optimization framework takes into account the effect of buildings on signal propagation, LAN lines and road topology. The formulation assumes a grid-based road network.

Trullols et al. (Trullols-Cruces et al., 2012) introduce a mixed-integer quadratic programming based optimal roadside units' deployment scheme to provide Internet access services for the maximum road traffic volumes with limited number of roadside units.

2.2.9 Strategies Based on Maximum Coverage

Trullols et al. (Trullols et al., 2010) study the placement of the roadside units into an urban area. The authors use a realistic data set and propose modeling the placement as a Knapsack Problem (KP) and also as a Maximum Coverage Problem (MCP-g). The heuristic MCP-g models the deployment of roadside units as a coverage problem. MCP-g assumes previously knowledge of all vehicles trajectories. Complementary, the KP heuristic does not assume knowledge of the vehicles trajectories.

Cataldi and Harri (Cataldi and Harri, 2011) propose the allocation of roadside units considering the Maximum Coverage Problem over a benefit function. The covered area is not a circle, but a polygon-based representing the measured heterogeneous any-directional communication conditions. Authors argue that the coverage area of an infrastructure node cannot be modeled as a circular shape because the intensity may not reflect the quality of the experienced connectivity. The benefit function is non-homogeneous over the covered area. The authors conclude on the existence of an upper bound on the number of roadside units that need to be allocate to some region.

Yongping et al. propose Roadgate (Xiong et al., 2013) to address the placement of roadside units guaranteeing a probability of contact between vehicles and the infrastructure. Xie et al. (Xie et al., 2013) address the placement of roadside units into a grid road network assuming knowledge of source and sink of each vehicle. Based on historical data the authors propose a probabilistic model to infer the better locations for the roadside units. The probabilistic model also uses a feature of the wireless link indicating the probability that a vehicle will get the information when driving through some roadside unit.

2.2.10 Mobility-Driven Deployment

Silva et al. (Silva et al., 2014b,a, 2015a; Silva and Meira Jr, 2015b,a; Silva et al., 2015b) observe that high mobility of nodes is probably the most distinguished aspect

of the vehicular networks, and such issue must be exploited by deployment strategies. Thus, the authors propose the use of the turning ratio at each intersection of the road network as a basic information to infer the mobility of vehicles. By using the turning ratios, authors define the concept of flow projection used to improve the allocation of the roadside units when we lack the vehicles trajectories.

2.2.11 Others

Rosi et al. (Rosi et al., 2008) propose an infrastructure to detect the position of a vehicle without the need of a GPS. Authors deploy a DSRC device at each intersection and they assume that vehicles have unique identifiers. In general terms they propose the aggregation of information to provide services such as the detection of congestion and emergency announcements. Lu et al. (Lu et al., 2013) investigate the capacity–cost trade-offs for vehicular access networks, in which access infrastructure is deployed to provide a downlink data pipe to all vehicles in the network. Three alternatives of wireless access infrastructure are considered: (i) cellular BSs; (ii) wireless mesh backbones (WMBs); and (iii) roadside access points (RAPs). Yan et al. (Yan et al., 2014) propose a class of algorithms named Tailor to select a minimum number of intersections to deploy the infrastructure using a 2-step approach. Liya et al. (Liya et al., 2013) propose a randomized algorithm that calculates an approximate distance for deploying roadside units by approaching the optimal distance step by step from the initial distance $d_0 = 2R_0$, where R_0 is the transmission range. Distance is sequentially increased to $d_0(1+\Theta)$, $d_0(1+\Theta)^2$, ..., $d_0(1+\Theta)^n$, until the network cannot meet the connectivity.

2.2.12 Remarks

Initial deployment proposals inherit assumptions from the cellular models. Typically, these works propose alternative ways to find the most crowded areas. Modern proposals consider mobility and they study the maximum achievable performance. In our opinion, research must now focus on real deployment scenarios without assuming tracking of individual vehicles.

2.3 Architectures

The ubiquitous connectivity provided by the cellular networks inspires the research community to investigate the benefits of integrating vehicles into a sophisticated

communication network. At the same time, Wi-Fi access points become more and more widespread in cities. While studies of ad-hoc vehicular communication keep focusing on radio devices, MAC protocols and routing protocols, a group of researchers visualizes the potential benefits of an infrastructure supporting vehicular communication (Wu et al., 2005a).

Important projects start to show relevant results, such as Fleetnet (Franz et al., 2001), Berkeley's California PATH[4] and CarTel (Bychkovsky et al., 2006). In particular, CarTel evaluates the vehicle-to-infrastructure communication with city-wide trials in Boston. CarTel reports the upload bandwidth to vehicles using the unplanned open residential access. Authors observe that the plethora of 802.11b access points spreading in cities can provide intermittent connectivity with high performance (when available).

Wu et al. (Wu et al., 2005a) states that infrastructure assessments are necessary to: (a) evaluate the communications architectures to identify those best suited for providing high bandwidth communications to travelers; (b) examine design options and trade-offs; and (c) quantitatively assess alternate approaches and evaluate their performance and reliability under realistic traffic conditions.

Figure 2.3 presents the number of published articles per year.

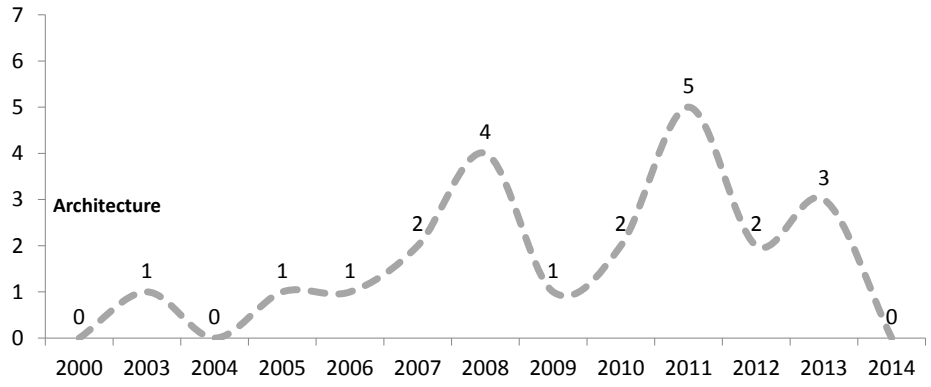


Figure 2.3. Articles presenting 'architectures' for infrastructure-based vehicular networks: x-axis indicates year, while y-axis indicates number of published works.

In 2002, Hsieh and Sivakumar (Hsieh and Sivakumar, 2002) classify the infrastructure into five categories:

- **WWAN last-hop:** Cellular-based WWAN is deployed to cover large areas. However, conventional cellular networks are unable to support a large number of high-bandwidth users;

[4]California PATH project. `http://www.path.berkeley.edu`

- **WLAN last-hop:** High-speed roadside WLAN access points are used. High throughput within the WLAN coverage area with much lower cost than WWAN is possible, but ubiquitous coverage requires many WLAN access points;
- **Multi-hop WLAN:** This is an extension of the previous design. Ad hoc vehicle-to-vehicle communications extend the reach of WLAN access points reducing the number of required access points, however, connectivity cannot be guaranteed due to the uncertainty of ad hoc communications;
- **WWAN last-hop + WLAN last-hop:** WWAN covers the entire area while WLAN access points are placed in hot spots. Where WWAN and WLAN overlap vehicles can choose to access either or both. Outside the WLAN coverage area, vehicles can only access the WWAN;
- **WWAN last-hop + Multi-hop WLAN:** Vehicles can access the WLAN either directly or through V2V communication. Vulnerability of multi-hop paths is mitigated because WWAN can provide connectivity when no path exists.

Figure 2.4 presents the distribution of works addressing architectures for infrastructure-based vehicular networks.

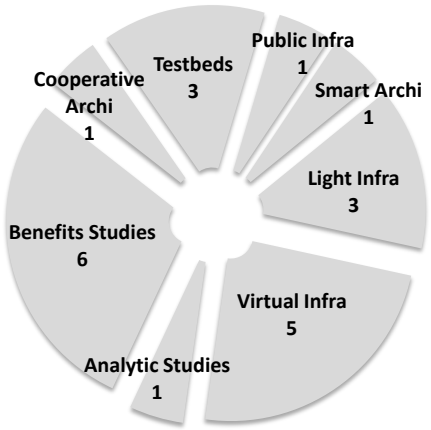


Figure 2.4. Organization of works addressing 'architectures' in infrastructure-based vehicular networks.

- **Analytic Studies**: efforts performing analytical investigations of a given property or characteristic of an architecture;
- **Benefits Studies**: efforts demonstrating the benefits of assisting vehicular networks within an infrastructure;

- **Cooperative Architectures**: efforts proposing cooperative/colaborative architectures;
- **Testbeds**: real implementation of vehicular networks assisted by infrastructure;
- **Public Infra**: efforts investigating the use of public infrastructure to assist vehicular communication;
- **Smart Architectures**: efforts proposing intelligent architectures;
- **Light Infra**: efforts proposing the use of less expensive devices to assist vehicular communication;
- **Virtual Infra**: efforts proposing new architectures where special nodes assist vehicular communication.

Sections 2.3.1 to 2.3.8 detail these works.

2.3.1 Analytic Studies

Abdrabou and Zhuang (Abdrabou and Zhuang, 2011) propose a framework based on queuing theory that gives a delay bound for relaying messages to the infrastructure through vehicle-to-vehicle communication. Authors study the multi-hop packet delivery delay in a low density vehicular network addressing a disrupted vehicle-to-infrastructure communication scenario where an end-to-end path is unlikely to exist between a vehicle and the nearest roadside unit.

2.3.2 Benefits of Incorporating an Infrastructure

Kozat and Tassiulas (Kozat and Tassiulas, 2003) consider the transport capacity of an ad hoc network with a random flat topology under the presence of an infinite capacity infrastructure network. The main idea is to use the nodes as relays and the infrastructure as pathways to drive the message as fast as possible to the destination node. Results demonstrate that a significant improvement can be achieved using the infrastructure support. Banerjee et al. (Banerjee et al., 2008) present a comprehensive comparison between different types of roadside units. Both analytical and simulation results reveal that relay and mesh nodes, as opposed to base stations, can be more cost-effective solutions even though a much larger number of such units are required to deliver the same level of performance as offered by the base stations. In addition, the

authors also suggest that adding a small amount of infrastructure is vastly superior to even a large number of mobile nodes capable of routing to one another.

Reis et al. (Reis et al., 2011) study the effects of including roadside units as relay nodes to improve communication in highway scenarios. Authors model the average time taken to propagate a packet to disconnected nodes when considering both scenarios of connected and disconnected roadside units. The trade-off between the required number of roadside units and the vehicular network performance in sparse scenarios is thus an important problem that requires careful study. Results show that significant improvements may be achieved when employing roadside units. For single-gap communications, the transmission delay may be reduced by 15% to 30%; for traversing multiple gaps, up to 25% reduction in end-to-end delay with disconnected roadside units is achievable, and with connected roadside units the decrease in delay can be of several orders of magnitude, depending on the road network. Authors conclude that sparse scenarios require a strong deployment of roadside units. In scenarios with multiple clusters, the connection between roadside units can greatly reduce the time to transmit information between vehicles. The use of roadside units to solve the disconnected network problem is still an important issue to tackle to be able to provide reliable communications for vehicular applications.

Gerla et al. (Gerla et al., 2006) propose **vehicular grid**, a large scale ad hoc network with ubiquitous presence of the infrastructure. Vehicular Grid must be entirely self-supporting for emergency operations (natural disaster, terrorist attack, etc), and it should also exploit the infrastructure during normal operations. Goal of the work is to bring Internet access to drivers. Authors argue that the access is possible because every vehicle will be only a few hops away from the infrastructure (Wi-Fi, cellular, etc.). Authors show that routing propagation can be done across the urban environment. Additionally, Gerla and Kleinrock (Gerla and Kleinrock, 2011) perform an historical comparison of the evolution of the Internet to identify the possible paths followed by the vehicular communication technology. Authors argue that the type of assistance requested from the infrastructure will vary according to application. Ideal access points' installations for vehicles are traffic lights, light poles, overpasses and other public structures. Traffic lights in particular are perfectly positioned to act as traffic routers. They already form a traffic grid; are located where traffic is most intense, and; are equipped with power and directly maintained by local municipalities. Among the lessons learned from the Internet evolution, multi-hop wireless networks are extremely inefficient due to collisions, lost packets, long paths, etc.

Mershad et al. (Mershad et al., 2012) propose **ROAMER**[5] to exploit roadside units to route packets between any source and destination in a vehicular network. Basic motivation behind using roadside units to route packets is that roadside units are stationary. It is much easier to send a packet to a fixed near target than to a remote moving object. ROAMER forwards packets to multiple neighbors to increase the chances of reaching destination without significantly increasing the overall traffic. Authors evaluate the roadside unit backbone routing performance via the ns2 simulation platform and demonstrate the feasibility and efficiency of the scheme in terms of query delay, packet success delivery ratio, and total traffic.

2.3.3 Cooperative Architectures

Liang and Zhuang (Liang and Zhuang, 2012) propose roadside wireless local area networks (RS-WLANs) as a network infrastructure for data dissemination. Authors propose a two-level cooperative data dissemination approach. For the network level, the aim is use available RS-WLANs for services to nomadic users. Packet level cooperation uses cooperative caching/transmission to improve the transmission rate: cooperative caching reduces the perception of limited bandwidth whereas cooperative transmission improves the packet transmission rate.

2.3.4 Light Architectures

Luan et al. (Luan et al., 2013) propose an infrastructure composed of roadside buffers, devices with limited buffer storage and wireless connection to support the vehicular communication with the goal to reduce the costs of network deployment. During 2007 the community also focuses on the shortness of contact time between vehicles and infrastructure, and it becomes clear that scheduling algorithms should consider the data size, deadline or even employ broadcasting to serve a large number of requests.

Zhang et al. (Zhang et al., 2007) propose a scheduling scheme for roadside units to provide a balance between serving downloads and upload requests from fast-moving vehicles on highways. Infrastructure acts as routers to Internet access. Although Internet connection demonstrates to be of great value for drivers, the deployment and maintenance costs of the infrastructure are considered very high. Thus, the authors propose the deployment of cheap roadside units acting as buffers between vehicles. Mishra et al. (Mishra et al., 2011) propose the use of stationary info-stations and moving vehicles in a publish-subscribe model. Vehicles can act as publishers,

[5]ROAdside Units as MEssage Routers in VANETs.

subscribers or brokers. Every major crossing of city is equipped with stationary info-stations that act as ultimate place holders for publications and subscriptions.

2.3.5 Publicly Available Infrastructure

Marfia et al. (Marfia et al., 2007) exploit the use of public Wi-Fi access points to provide vehicular communication. Authors map the public access points available in the city of Portland (US) and vehicles can opportunistically use the infrastructure to communicate with other vehicles in order to avoid long wireless ad hoc paths, and to alleviate congestion in the wireless grid. Analytic and simulation models are used to optimize the communications and networking strategies. Authors conclude that the motion model has enormous impact on the results and that the presence of infrastructure largely improve the communication.

2.3.6 Smart Architectures

Infrastructure for gaming over vehicular networks is addressed by Palazzi et al. (Palazzi et al., 2008, 2009). Authors investigate the problematic coexistence between TCP and UDP flows in the context of infrastructure-based vehicular networks. They observe that the retransmissions of TCP are exacerbated in vehicular networks since the high mobility of the vehicles generates continuous variations in the number and type of flows served by the infrastructure along the road. Thus, they propose the use of smart access points along roads, able to regulate heterogeneous transmission flows and make them coexist efficiently. Basically, smart access points snoop transiting packets of various flows and computes the maximum data rate at which each elastic application will be able to transfer data without incurring in congestion losses. Data rate is computed and also included on-the-fly in transiting ACKs of TCP flows. They validate their strategy using the ns-2 simulator. Authors use a grid-based road network streaming the video Star Wars IV in high quality MPEG4 format. Online gaming traffic is inspired by real traces of the popular Counter Strike action game, and it has: (a) a server-to-client flow characterized by an inter-departing time of game updates of 200 bytes every 50ms; and (b) a client-to-server flow of 42 bytes every 60ms.

2.3.7 Testbeds

Wu et al. (Wu et al., 2005b) propose a real test-bed for evaluating communication between moving vehicles and infrastructure. Authors exploit the communication between vehicles and the infrastructure. Opportunistic forwarding

(store-carry-forward) appears to be a viable approach for data dissemination using vehicle-to-vehicle communications for applications that can tolerate some data loss and delay. Studies show vehicle-to-vehicle communication is feasible, although the propagation performance depends on factors such as the density of instrumented vehicles along the end-to-end path.

Authors propose the infrastructure to reduce path vulnerability in critical areas or in a subset of vehicles equipped with cellular messaging systems. Field experiments were conducted using a laptop, 802.11b[6] card with a 2.5dB omni-directional external antenna placed on the roof of the vehicle, and a GPS receiver. They measured the wireless communication performance between a fixed roadside station and a moving vehicle. Most of our measurements show more than 500m of effective communication range. They also measured communication performance between two vehicles traveling in opposite directions. Most test cases showed more than 200m of effective communication range. Average time for effective communication is about 21s. Authors conclude that vehicular communication is feasible.

Ormont et al. (Ormont et al., 2008) mounted a testbed in the city of Madison, Wisconsin to monitor Wi-Fi[7] signals over the city. Main communication channel is the 3G cellular network. Clients were installed in two buses. Each city bus operates on multiple routes on a single day and is, therefore, able to traverse through significant parts of the city. Buses provide Internet access to passengers through 3G connection. Client is a laptop with a Wi-Fi interface running software that monitors and stores Wi-Fi networks found. Cellular interface provides continuous remote access to each testbed node to experimenters. Client uses it to periodically upload measurement data to a backend database. Authors argue that such testbed can be used to draw coverage maps, analyze performance at specific locations, infer mobility patterns and study relationships between performance and mobility.

Fernandez Ruiz et al. (Fernandez Ruiz et al., 2010) study the handover using WiMAX[8] and Wi-Fi applied to vehicular communication. They have mounted a testbed in the Campus of Espinardo, University of Murcia. Campus has a ring road that surrounds a huge enough building area. Any vehicle connected to the wireless network can freely move, using different access points that could be available throughout its path. These access points could belong to different domains and different wireless technologies like Wi-Fi, WiMAX and Universal

[6]IEEE 802.11 Wireless Local Area Networks Working Group. `http://www.ieee802.org/11/`
[7]IEEE 802.11. `http://standards.ieee.org/getieee802/802.11.html`
[8]IEEE 802.16. `http://wirelessman.org/`

Mobile Telecommunication System (UMTS)[9]. As a consequence of this, several types of handovers can be differentiated: (i) Intra-domain intra-technology handover; (ii) Intra-domain inter-technology handover; (iii) Inter-domain intra-technology handover; (iv) Inter-domain inter-technology handover.

Authors use Mobile Internet Protocol for IPv6 (MIPv6)[10] in order to make the vehicles change from service providers but keeping the same IP address. They conclude that the deployment of wireless infrastructures must take into account the surrounding environment and the specific circumstances, using the advantages of the different wireless technologies available transparently to the end user.

2.3.8 Virtual Infrastructure

Jerbi et al. (Jerbi et al., 2008) observe that the need for an infrastructure can decrease the area of vehicular network applications, so the authors propose a self-organizing mechanism to emulate a geo-localized virtual infrastructure in order to avoid the costs of deployment. Authors use vehicles currently populating the geographic region. Geo-localized virtual infrastructure mechanism consists on electing vehicles that will perpetuate information broadcasting within the intersection area. Geo-localized virtual infrastructure is composed of two phases: (i) Select vehicles able to reach the broadcast area; (ii) Only one among the selected vehicles is elected as the local broadcaster. Elected vehicle performs a local/single hop broadcast once it reaches the broadcast area. Authors conclude that the proposed geo-localized virtual infrastructure can: (i) periodically disseminate the data within a given area; (ii) efficiently utilize the limited bandwidth; (iii) ensure a high delivery ratio.

Luo et al. (Luo et al., 2010) propose MI-VANET[11], a two-tier architecture: buses constitute the mobile backbone for data delivery while the low tier is composed of ordinary cars and passengers. Cars must register in buses in order to send/receive data. There is a score mechanism to choose the best bus. When the car is leaving the communication range of its registered bus, another bus will be chosen for registration. Authors use VanetMobiSim as a traffic simulator, and they assume that: (i) vehicles are uniformly distributed over the road; and (ii) buses represent 20% of the vehicles. Routing algorithm used on the high tier is called Mobile Infrastructure Routing. Each bus knows its location and has a digital street map including bus line information. MIRT is a location based reactive routing protocol that selects the optimal route and

[9]UMTS-Universal Mobile Telecommunications System. `http://www.protocols.com/pbook/umts.htm`

[10]Mobile IPv6 Home Page. `http://mobile-ipv6.org`

[11]Mobile Infrastructure Based VANET.

forwards the request hop-by-hop. Simulation results show that there is a 40-55% improvement in delivery ratio while the throughput is even doubled compared to Greedy perimeter stateless routing for wireless networks (GPSR)[12] in traditional vehicular networks.

Annese et al. (Annese et al., 2011) study the communication of vehicles to infrastructure to provide UDP-based multimedia streams. Work considers continuous coverage of infrastructure within the urban road topology and analyzes the vehicular communication as a Mesh (Akyildiz et al., 2005) network. Mesh networks are typically free-standing robust systems that can be conveniently integrated with existing infrastructure and offer high bit rate. Authors do not assume vehicles as end nodes like proposed in works (Capone et al., 2007; Amir et al., 2006; Ramani and Savage, 2005; Brik et al., 2005), but as mesh nodes connecting the wireless medium and acting as routers. They argue that such new point of view is important because it allows the routing protocol to run on the mobile node itself, better adapting to the high-mobility profile of the node. Vehicle becomes a mobile hot spot that can act as a mesh gateway.

Because of high investments required to deploy roadside units over large areas, Tonguz and Viriyasitavat (Tonguz and Viriyasitavat, 2013) propose an alternative approach to roadside infrastructure by leveraging the use of existing DSRC-equipped vehicles to provide roadside unit functionality. Approach employs a self-organizing network paradigm and draws its inspiration from social biological colonies such as ants, bees, birds, and fishes. Such an approach was initially formulated in (Tonguz, 2011). Vehicles that act as temporary roadside units can make brief stops during which they act as a communication bridge for other vehicles in the network. Each vehicle runs the distributed gift-wrapping algorithm proposed by (Viriyasitavat et al., 2010). Upon receiving a message the vehicle determines whether it lies on the boundary of a coverage polygon. As a drawback, vehicles acting as temporary roadside units need to make brief stops (approximately 30s) to reach the maximum number of uninformed vehicles. Authors argue that such increase in travel time is small when compared to increases due to accident-induced congestion.

Sommer et al. (Sommer et al., 2013) propose utilizing parked vehicles as relay nodes to address the disconnected network problem. Extensive simulations and real life experiments show that parked cars can increase cooperative awareness by over 40%.

[12]GPSR-Greedy Perimeter Stateless Routing. `http://www.cs.cmu.edu/~./bkarp/gpsr/gpsr.html`

2.3.9 Remarks

Research on architectures is initially motivated by the need to offer Internet access to vehicles. Several kinds of architectures are proposed, including the use of public Wi-Fi access points to enhance the vehicular communication. Infrastructure gains visibility as the researchers realize its importance to support the dissemination of data in vehicular networks. For a matter of costs[13], some researches focus on alternative and less-expensive methods to achieve the benefits of a dedicated infrastructure. Smart architectures and light architectures are also proposed.

2.4 Communication

This section presents efforts addressing communication in infrastructure-based vehicular networks. Evolution of articles per year is indicated in Figure 2.5.

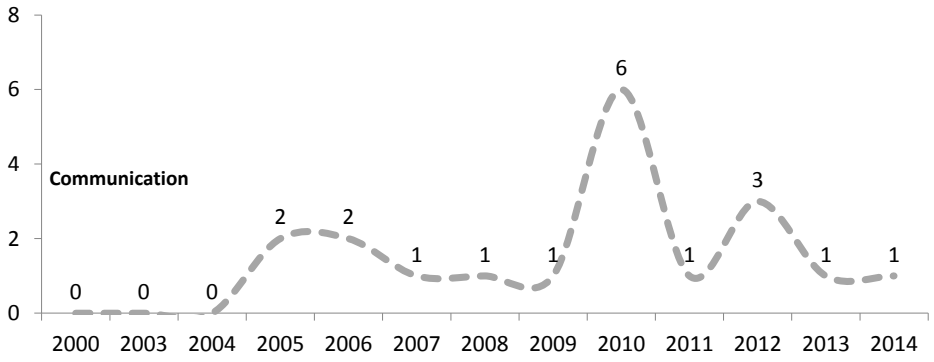


Figure 2.5. Number of published articles addressing 'communication' in infrastructure-based vehicular networks: x-axis indicates year, while y-axis indicates number of published works.

Works addressing communication in infrastructure-based vehicular networks can be classified according to Figure 2.6.

Sections 2.4.1 to 2.4.7 detail these works.

2.4.1 Analytic Studies

Abdrabou and Zhuang (Abdrabou and Zhuang, 2009) propose an analytical framework that helps to approximately estimate the minimum number of roadside units required to cover a road segment with a probabilistic vehicle-to-infrastructure delay guarantee, given that an intermittent multi-hop connectivity exists between vehicles and roadside

[13]Unless state otherwise, we are assuming monetary cost.

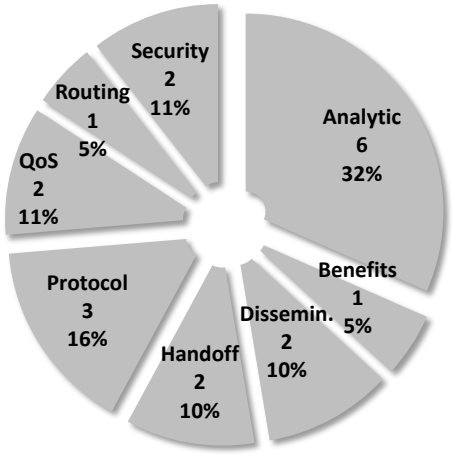


Figure 2.6. Organization of works addressing 'communication' in infrastructure-based vehicular networks.

units and vehicles are sending burst traffic. In another work, (Abdrabou et al., 2010) present a study of the relation between packet delivery delay and roadside unit density for vehicular-to-infrastructure communication in sparse vehicular ad hoc networks where vehicles moving in one direction send their packets to vehicles traveling in the opposite direction in order to deliver the packets to the nearest roadside unit.

Chun et al. (Chun et al., 2010) study the access probability considering an infrastructure wherein a number of base stations are uniformly deployed along a long road, while other vehicles or cars are distributed on the road according to a Poisson distribution. Authors formulate a mathematical model that relates the density of vehicles, coverage range and distance between adjacent base stations to infer the access probability.

Chun and Mao (Chun and Mao, 2010) analyze the probability of k-hops connectivity in infrastructured wireless networks, while (Malandrino et al., 2011) address the content download in vehicular networks leveraging both infrastructure-to-vehicle and vehicle-to-vehicle communication. Authors formulate a max-flow problem that accounts for practical aspects including channel contention and the data transfer paradigm. The goal is to answer "what is the maximum downloading performance theoretically achievable through DSRC-based I2V/V2V communication in a given mobility scenario?" Authors conclude that: (i) density-based Access Points deployment yields close-to-optimum performance, and (ii) multi-hop traffic delivery is beneficial, although the gain is negligible beyond two hops from the Access Point.

Zhang et al. (Zhang et al., 2012) propose an analytical model to predict both uplink and downlink connectivity probabilities. The uplink connectivity probability is defined as the probability that messages from vehicles can be received by the

infrastructure through multi-hop paths. The downlink connectivity probability is defined as the probability that messages can be broadcasted from roadside units to all vehicles through multi-hop paths. Analytical results reveal the trade-off between both performance metrics.

2.4.2 Infrastructure Benefits in Communications

Lochert et al. (Lochert et al., 2007) discuss the initial moments of a vehicular network and the authors demonstrate that during the rollout phase some kind of support is needed. Otherwise, many envisioned applications are unlikely to work until a large fraction of vehicles participate. Authors use stationary support units to improve the refreshing rate of the information dissemination in city scenarios.

2.4.3 Infrastructure-based Data Dissemination

Liu and Lee (Liu and Lee, 2010) present a data dissemination strategy. In heavy traffic conditions, the authors propose the push-based broadcast data dissemination: messages are periodically broadcasted to passing vehicles. In light traffic scenarios, the vehicles query on-demand for traffic information. The authors derive a mathematical model that shows the effectiveness of their solution and they conclude that data dissemination in vehicular networks should be adaptable to dynamic traffic environments: dynamic channel and data allocation is a critical, but an effective mechanism in providing hybrid scheduling between push-based and on-demand services.

Bruno and Nurchis (Bruno and Nurchis, 2013) assume vehicles equipped with cameras and the problem is how to deliver the images to remote data collectors. Authors propose a data collection algorithm capable of eliminating the redundancy of data transmitted by moving vehicles. In a real situation, several vehicles may report the same event. Thus, the data redundancy mitigation is necessary to improve the network efficiency. The model is based on a maximum coverage problem followed by a sub modular optimization.

2.4.4 Quality of Service

Quality of service is considered in two works: Luan et al. (Luan et al., 2012) focus on the MAC layer for V2I communications where multiple fast moving vehicles with different on-top applications and QoS requirements compete for the transmissions to the road-side infrastructure. Complementary, Harigovindan et al. (Harigovindan et al., 2014) develops a mechanism for fair channel allocation.

2.4.5 Network Handoff

Network hand-off is also addressed in two works: Ramani and Savage (Ramani and Savage, 2005) propose SyncScan to continuously track nearby base stations by synchronizing short listening periods at the client with periodic transmissions from each base station. Brik et al. (Brik et al., 2005) propose MultiScan so that nodes rely on using their (potentially idle) second wireless interface to opportunistically scan and pre-associate with alternate access points and eventually seamlessly handoff ongoing connections.

2.4.6 Protocols for Infrastructure-based Vehicular Networks

Hadaller et al. (Hadaller et al., 2006) propose MV-MAX (Multi-Vehicular Maximum), a protocol to increase the global data transfer. Authors observe that when a roadside access point is shared by more than one vehicle, the vehicle with the lowest transmission rate reduces the effective transmission rate of all other vehicles. Observing that every vehicle eventually receives good performance when it is near the access point, the authors propose a medium access protocol that opportunistically grants access to vehicles with maximum transmission rate. The overall system throughput is improved by a factor of four.

Korkmaz et al. (Korkmaz et al., 2006) propose a cross-Layer multi-hop data delivery protocol with fairness guarantees where vehicles do not communicate with roadside units individually, but through one leader. The goal is to reduce the network traffic and to use bandwidth more efficiently. The leader will collect all information from other nodes and share it with roadside units. Additionally, Korkmaz et al. (Korkmaz et al., 2010) propose a new protocol that employs fixed gateways along the road which perform periodic admission control and scheduling decisions for the packet traffic in their service area. The most important contribution of the protocol is providing delay bounded throughput guarantees for soft real-time traffic, which is an important challenge especially for a mobile multi-hop network. After the demands of the soft real-time traffic are met, the protocol supports the best-effort traffic using remaining bandwidth.

2.4.7 Routing in Infrastructure-based Vehicular Networks

Borsetti and Gozalvez (Borsetti and Gozalvez, 2010) propose an infrastructure-assisted routing approach designed to improve the end-to-end performance, range and operation of multi-hop vehicular communications by exploiting the reliable interconnection of

infrastructure units. The infrastructure is wired-connected and uses the position of each vehicle to route the data. Authors use SUMO traces and a grid streets layout. They conclude that to obtain the maximum benefits from the proposed infrastructure-assisted routing approach, optimal infrastructure deployment strategies must be further investigated.

2.4.8 Security

Plobl and Federrath (2008) propose a set of security requirements for the infrastructure of vehicular networks in terms of integrity, confidentiality and availability. In order to protect the integrity, the security infrastructure has to provide mechanisms that prevent and detect the modification of messages. Furthermore, the authenticity and integrity of the message must be provable instantly without further information. Proof of integrity and origin of data is recommended to prevent misuse of the network combined with correct time and position information in all messages to protect against replay and position spoofing attacks.

In terms of confidentiality, the security infrastructure has to provide mechanisms that support different levels of confidentiality, and all messages should be protected against eavesdropping. In terms of availability, the network must provide real-time processing of messages, possibly using data compression techniques to reduce network bandwidth consumption combined with actions to complicate denial-of-service attacks. Authors propose that after a once-only initialization, the system employs asymmetric cryptography within a public key infrastructure for messages influencing road safety. All other messages are protected by a system employing symmetric cryptography.

Marmol and Perez (Marmol and Perez, 2012) propose TRIP[14], a model used to decide whether to accept a traffic warning coming from other vehicle or not by assessing the trustworthiness of the issuer of such message. Authors extend the requirements of the vehicular infrastructure by proposing that the security infrastructure should also be able to make fast decisions to deal with the constantly changing topology and fast switching of neighbors; otherwise, the communication becomes very inefficient. Network should also be resilient to security and privacy threats such as malicious nodes trying to drive the reputation of a reliable node down. Finally, security must also be independent of mobility patterns in order to accurate perform under every possible traffic scenario.

[14]Trust and Reputation Infrastructure-based Proposal.

2.5 Physical Communication: 4G or DSRC

Any vehicle may possess several communication interfaces, and the selection of the communication interface may consider specific application needs and minimum QoS metrics. Throughout this thesis we address how to position a minimum infrastructure supporting the vehicular network. Across the text we have discarded any technology-specific assumptions or details related to the physical communication itself, and we have turned our attention to locate the infrastructure according to the vehicular mobility. In some sense we have assumed the physical proximity as a decision criterion for the establishment of the communication, since we assume physical proximity, in some sense we are also assuming the DSRC as our communication technology.

However, vehicles composing the network are not supposed to rely on a single communication interface. As the demand for Internet connectivity rises, existing cellular hardware may reduce the costs for integrating communication into vehicles, and broadly adopted vehicular applications demonstrate the feasibility of leveraging the communication into vehicles by considering the cellular systems.

Both DSRC and cellular systems present interesting characteristics for the vehicular communication. The DSRC performs well when we consider short distances up to a few hundred meters between communicating pairs, and also demonstrates a lower transmission delay (physical proximity). A drawback of the DSRC is the Non-Line-of-Sight issue, which may be mitigated by using roadside units or vehicles (if available) to redirect the signal.

On the other hand, cellular systems are spread over the cities providing ubiquitous connectivity, and we can rely on such pervasive infrastructure to increase the network participation. For instance, any vehicle owning a 4G connection may act as a virtual roadside unit by offering gateway functionalities between the DSRC and the cellular network, collecting traffic information and disseminating them to near vehicles. Gerla and Kleinrock (2011) discusses the future of the vehicular applications, and the authors consider that most of the applications will be designed in a peer-to-peer (P2P) basis. However, a large amount of communicating vehicles within a given cell may overload the cellular systems.

The fact is that DSRC and 4G are likely to coexist. Every communication has a given purpose. And the purpose of the communication is defined by the applications running on top of it. And each application has its own communication requirements. While some applications will benefit from the cellular systems, others will demand the DSRC communication. For instance, Vinel (2012) compare the performance of DSRC and 4G and they conclude that 4G is not likely to be able to provide the same awareness

update rate and latency as the DSRC. On the other hand, DSRC is not able to provide all the pervasiveness provided by the cellular systems.

2.6 Remarks

This work presents an overview of 68 published articles proposing solutions for infrastructure-based vehicular networks in order to identify the main questions addressed by the research community from Jan. 2003 to Feb. 2014. We use the formalism of a Systematic Literature Review in order to make our selection impartial. Our data sources are: ACM, IEEE and Elsevier. For the purposes of this chapter, we have organized selected works into three classes: architecture, communication, and deployment. In particular, class 'deployment' groups works proposing strategies to position the infrastructure in order to maximize communication benefits in vehicular networks. Deployment leads the number of published articles since 2012.

2.7 Overview of the Next Chapter

In the next chapter we develop our PMCP-based deployment algorithm. We use a MCP-based approach as our baseline (also detailed in next chapter). Throughout the chapter we assume a simple and theoretical road network, and we perform several experiments to evaluate our proposal.

2.8 List of Selected Articles

- Table 2.3 indicates articles classified as **architecture**;
- Table 2.1 indicates articles classified as **deployment**;
- Table 2.2 indicates the articles classified as **communication**.

Table 2.1. Articles of Deployment of Infrastructure for Vehicular Networks

Article	Reference
Optimal Placement of Gateways in Vehicular Networks	(Li et al., 2007)
The capacity of Vehicular Ad Hoc Networks with infrastructure	(Nekoui et al., 2008)
A Novel Approach for Infrastructure Deployment for VANET	(Rosi et al., 2008)
Data aggregation and roadside unit placement for a vanet traffic information system	(Lochert et al., 2008)
Alpha coverage: bounding the interconnection gap for vehicular Internet access	(Zheng et al., 2009)
Centrality-based Access-Points deployment for vehicular networks	(Kchiche and Kamoun, 2010)
Maximizing the contact opportunity for vehicular Internet access	(Zheng et al., 2010)
Planning roadside infrastructure for information dissemination in intelligent transportation systems	(Trullols et al., 2010)
A Power-Saving Model for Roadside Unit Deployment in Vehicular Networks	(Sou, 2010)
Enhancing VANET Connectivity Through Roadside Units on Highways	(Sou and Tonguz, 2011)
User/Operator Utility-Based Infrastructure Deployment Strategies for Vehicular Networks	(Cataldi and Harri, 2011)
Non-cooperative RSU deployment in vehicular networks	(Filippini et al., 2012)
Cooperative download in vehicular environments	(Trullols-Cruces et al., 2012)
Roadside Unit Deployment for Information Dissemination in a VANET : An Evolutionary Approach	(Cavalcante et al., 2012)
Optimal roadside units placement in urban areas for vehicular networks	(Aslam et al., 2012)
Optimal Placement and Configuration of Roadside Units in Vehicular Networks	(Liang et al., 2012)
A Cost-Effective Strategy for Road-Side Unit Placement in Vehicular Networks	(Wu et al., 2012a)
Intersection-priority based optimal RSU allocation for VANET	(Chi et al., 2013)
A Randomized Algorithm for Roadside Units Placement in Vehicular Ad Hoc Network	(Liya et al., 2013)
File downloading oriented Roadside Units deployment for vehicular networks	(Liu et al., 2013)
Vehicles Meet Infrastructure: Toward Capacity–Cost Tradeoffs for Vehicular Access Networks	(Lu et al., 2013)
A geometry-based coverage strategy over urban VANETs	(Cheng et al., 2013)
Voronoi-based placement of road-side units to improve dynamic resource management in Vehicular Ad Hoc Networks	(Patil and Gokhale, 2013)
Access Points Planning in Urban Area for Data Dissemination to Drivers	(Yan et al., 2014)
Design of Roadside Infrastructure for Information Dissemination in Vehicular Networks	(Silva et al., 2014a)

Table 2.2. Articles of Communication for Infrastructure-Based Vehicular Networks

Article	Reference
Eliminating Handoff Latencies in 802.11 WLANs Using Multiple Radios:	(Brik et al., 2005)
SyncScan: Practical Fast Handoff for 802.11 Infrastructure Networks	(Ramani and Savage, 2005)
MV-MAX: improving wireless infrastructure access for multi-vehicular comm.	(Hadaller et al., 2006)
A cross-layer multihop data delivery protocol with fairness guarantees for vehicular networks	(Korkmaz et al., 2006)
The feasibility of information dissemination in vehicular ad hoc networks	(Lochert et al., 2007)
On a Stochastic Delay Bound for Disrupted Vehicle-to-Infrastructure Communication with Random Traffic	(Abdrabou and Zhuang, 2009)
Analysis of Access and Connectivity Probabilities in Infrastructure-Based Vehicular Relay Networks	(Chun et al., 2010)
Delay Analysis for a Reliable Message Delivery in Sparse Vehicular Ad Hoc Networks	(Abdrabou et al., 2010)
Analysis of k-Hop Connectivity Probability in 2-D Wireless Networks with Infrastructure Support	(Chun and Mao, 2010)
RSU-based real-time data access in dynamic vehicular networks	(Liu and Lee, 2010)
Supporting real-time traffic in multihop vehicle-to-infrastructure networks	(Korkmaz et al., 2010)
Infrastructure-assisted geo-routing for cooperative vehicular networks	(Borsetti and Gozalvez, 2010)
Content downloading in vehicular networks: What really matters	(Malandrino et al., 2011)
Multi-Hop Connectivity Probability in Infrastructure-Based Vehicular Networks	(Zhang et al., 2012)
Provisioning QoS controlled media access in vehicular to infrastructure comm.	(Luan et al., 2012)
Robust and efficient data collection schemes for vehicular multimedia sensor Networks	(Bruno and Nurchis, 2013)
Proportional fair resource allocation in vehicle-to-infrastructure networks for drive-thru Internet applications	(Harigovindan et al., 2014)
A privacy aware and efficient security infrastructure for vehicular ad hoc networks	(Plobl and Federrath, 2008)
TRIP, a trust and reputation infrastructure-based proposal for vehicular ad hoc networks	(Marmol and Perez, 2012)

Table 2.3. Articles of Architectures for Infrastructured Vehicular Communication.

Article	Reference
The infostations challenge: Balancing cost and ubiquity in delivering wireless data	(Frenkiel et al., 2000)
Throughput capacity of random ad hoc networks with infrastructure support	(Kozat and Tassiulas, 2003)
Proportional fair resource allocation in vehicle-to-infrastructure networks for drive-thru Internet applications	(Ott and Kutscher, 2004)
An architecture study of infrastructure-based vehicular networks	(Wu et al., 2005a)
Vehicular networks in urban transportation systems	(Wu et al., 2005b)
Vehicular Grid Communications: The Role of the Internet Infrastructure	(Gerla et al., 2006)
The CarTel Mobile Sensor Computing System	(Bychkovsky et al., 2006)
Evaluating vehicle network strategies for downtown Portland	(Marfia et al., 2007)
On Scheduling Vehicle-Roadside Data Access	(Zhang et al., 2007)
Relays, base stations, and meshes: Enhancing mobile networks with infrastructure	(Banerjee et al., 2008)
How to let gamers play in infrastructure-based vehicular networks	(Palazzi et al., 2008)
A city-wide vehicular infrastructure for wide-area wireless experimentation	(Ormont et al., 2008)
Geo-localized virtual infrastructure for urban vehicular networks	(Jerbi et al., 2008)
Deployment of a Secure Wireless Infrastructure Oriented to Vehicular Networks	(Fernandez Ruiz et al., 2010)
MI-VANET: A New Mobile Infrastructure Based VANET Architecture for Urban Environment	(Luo et al., 2010)
Probabilistic Delay Control and Road Side Unit Placement for Vehicular Ad Hoc Networks with Disrupted Connectivity	(Abdrabou and Zhuang, 2011)
On the Performance of Sparse Vehicular Networks with Road Side Units	(Reis et al., 2011)
A Publish/Subscribe Communication Infrastructure for VANET Applications	(Mishra et al., 2011)
Seamless Connectivity and Routing in Vehicular Networks with Infrastructure	(Annese et al., 2011)
ROAMER: Roadside Units as message routers in VANETs	(Mershad et al., 2012)
Cooperative data dissemination via roadside WLANs	(Liang and Zhuang, 2012)
Engineering a distributed infrastructure for large-scale cost-effective content dissemination over urban vehicular networks	(Luan et al., 2013)
Cars as roadside units: a self-organizing network solution	(Tonguz and Viriyasitavat, 2013)
IVC in Cities: Signal Attenuation by Buildings and How Parked Cars Can Improve the Situation	(Sommer et al., 2013)

Chapter 3

Grid-Based Road Network

In this chapter we develop our PMCP-based deployment algorithm. We use a theoretical grid-based road network in order to make the formulation as intuitive as possible. In this formulation we assume roadside units always placed at roads intersections. The literature is plenty of proposals for vehicular applications offering a large spectrum of traffic information solutions. For instance: (a) Rybick et al. (2007) propose Peer-On-Wheels, a peer-to-peer solution for monitoring traffic conditions; (b) Eriksson et al. (2008) propose Pothole to monitor roads conditions; (c) Waze[1] (2008) proposes a collaborative solution for monitoring traffic conditions; (d) Smaldone et al. (2008) propose RoadSpeak for collaborative driving and chatting; (e) Thompson et al. (2010) propose a solution for the detection of events; (f) Zaldivar et al. (2011) propose a solution for detection of accidents; (g) Johnson and Trivedi (2011) propose an application for monitoring performance of vehicles; (h) Koukoumidis et al. (2011) propose SignalGuru, an application for recording the traffic lights schedule. Monitoring traffic lights is also proposed by Le et al. (2011) and Cai et al. (2010); (i) Araujo et al. (2012) propose monitoring drivers behavior for maximum efficiency of vehicles; (j) Silva et al. (2015c) proposes an application for non-stop traversal of low traffic intersections.

While these applications may work on an ad-hoc basis, a minimum infrastructure supporting these applications increases up to five times the message delivery ratio, and reduces up to 35% the expected delivery time (Kozat and Tassiulas, 2003; Gerla et al., 2006; Banerjee et al., 2008; Reis et al., 2011; Mershad et al., 2012; Wu et al., 2012b). Such relevant gains are explained by the intrinsic nature of the vehicular networks composed of dynamic nodes moving at high speeds changing the network topology. When considering rural areas and roadways, the gains are even stronger since the nodes are very disperse. The deployment of infrastructure for vehicular networks

[1]WAZE Mobile App. `http://www.waze.com`

presents several important issues. In this chapter we focus on a specific one:

> **Proposal of a deployment algorithm that considers partial mobility information aiming to maximize the number trips experiencing at least one V2I contact.**

Here we propose a deployment algorithm based on the Probabilistic Maximum Coverage Problem to compute the location of the roadside units within a road network (Silva et al., 2014a,b). Our approach considers the flow of vehicles within the road network in order to select the most promising intersections to receive the roadside units. Thus, we model the allocation of roadside units as a Probabilistic Maximum Coverage Problem. In our initial formulation, each intersection represents a set, each vehicle represents an element, and each element has a probability p of belonging to a given set. Our goal is to find the sets maximizing the expected cardinality of the union of the selected sets, i.e., the expected number of vehicles contacting the infrastructure. Thus, the Algorithm PMCP considers the flow of vehicles in order to select those intersections maximizing the number of trips experiencing at least one V2I contact. In the following chapters we remove the restriction of placing roadside units at intersections. But, for now let's assume that:

(a) Intersections are the ideal places to receive the roadside units;

(b) Transmitted information is a small self-contained item;

(c) Concept of infostations proposed by Frenkiel et al. (2000): vehicles do not necessarily require continuous coverage. Instead, vehicles engage in "infofueling" as they opportunistically drive past roadside access points.

We investigate the benefits of the probabilistic model by comparing a PMCP-based approach and a MCP-based one. However, since we do not assume previous knowledge of the vehicles trajectories, we are not truly able to solve PMCP nor MCP. Instead of it, we propose constructive heuristics to approximate both problems solutions. In order to investigate the benefits of incorporating partial mobility information, we develop both heuristics in a very similar basis, and the only distinction between them is the partial mobility information used by our PMCP-based approach. Our main findings are:

- A PMCP-based approach increases the number of V2I contacts when compared to a MCP-based approach;

- A PMCP-based approach requires less roadside units to achieve similar V2I contacts than a MCP-based one;
- A PMCP-based approach distributes the roadside units in a layout better fitting the traffic flow;
- A PMCP-based approach offers more regularity than a MCP-based approach in the number of V2I contacts experienced by vehicles.

Notice that we are assuming a slightly different concept of 'coverage': the traditional usage of coverage indicates a continuous region where users have a high probability of meeting connection. But, since we assume **infostations**, we consider small islands of coverage: fragmented and possibly disconnect areas where users are supposed to meet connection. The Probabilistic Maximum Coverage Problem considers the likelihood of a given vehicle to reach the intersection i_2 given that such vehicle is located at intersection i_1, which leads to a more complete evaluation of the vehicular mobility, allowing our deployment algorithm to select better intersections to receive the roadside units. In an abstract sense, Probabilistic Maximum Coverage Problem enables us to project the flow of vehicles within the road network, and this will be discussed along this work.

The main contribution of this chapter is to evaluate the performance gain achieved when we incorporate partial mobility information into a deployment algorithm. The remainder of this chapter is organized as follows. Section 3.1 details Maximum Coverage Problem. Section 3.2 presents Probabilistic Maximum Coverage Problem. Section 3.3 presents an analytic formulation of the theoretical grid. Section 3.4 formulates the roadside allocation problem. Section 3.5 discusses the simulations and results. Section 3.6 concludes the chapter.

3.1 Maximum Coverage Problem

Trullols et al. (2010) model the allocation of roadside units as a Maximum Coverage Problem. Maximum Coverage Problem is stated as follows:

Definition 1 (Maximum Coverage Problem) *Suppose a collection of sets* $S = \{S_1, S_2, ..., S_m\}$ *defined over a domain of elements* $X = \{x_1, x_2, ..., x_n\}$. *Repetition of elements is allowed. Goal is to find a collection of sets* $S' \subseteq S$ *such that the number of covered elements* $| \cup_{x_i \in X'} |$ *is maximized.*

Although Maximum Coverage Problem is NP-Hard (Cormen et al., 2001), it is well-known that the greedy heuristic achieves an approximation factor $(1 - 1/m)^m$, where m is the maximum cardinality of the sets in the optimization domain (Trullols et al., 2010). However, the greedy heuristic requires previous knowledge of the vehicles trajectories. Algorithm 1 presents MCP-g, the greedy solution for the Maximum Coverage Problem.

G**:** Vehicles trajectories information;

α**:** Number of available roadside units.

Algorithm 1 MCP-g: Greedy Solution (requires knowledge of trajectories).

Input: G, α;
Output: Γ (locations to receive RSUs);
1: $\Gamma \leftarrow \emptyset$; ▷ solution set starts empty
2: **for** i=1 to α **do**
3: $\Gamma \leftarrow \Gamma \cup$ location with max uncovered vehicles(G);
4: **end for**
5: **return** Γ;

The heuristic MCP-g chooses sets (i.e., locations) according to one rule: at each stage choose a set which contains the largest number of uncovered elements. MCP-g requires previous knowledge of the vehicles trajectories (full mobility information). Trullols et al. (2010) present an in-depth discussion of MCP-g. Tracking individual vehicles raises several privacy (and practical) issues, which has motivated us to perform the investigation presented in this work. Notice that when we remove the trajectories information, MCP-g has no means to identify the locations accounting for the maximum number of uncovered vehicles. Without the trajectories information, our only choice viable choice is to select the densest urban locations to receive the roadside units. We use MCP-based as an approximate solution for the Maximum Coverage Problem when we do not have access to the vehicles trajectories.

Algorithm 2 presents MCP-based.

M**:** Matrix describing the concentration of vehicles within the target region;

α**:** Number of available roadside units.

MCP-based receives the matrix of locations (M) where each $M_{i,j}$ stores the number of vehicles that have crossed location $\{i, j\}$, and the number of available roadside units (α) to be deployed. MCP-based selects (at each iteration) the location

Algorithm 2 MCP-based (without knowledge of trajectories).

Input: M, α;
Output: Γ (locations to receive RSUs);
1: $\Gamma \leftarrow \emptyset$; ▷ solution set starts empty
2: $\vartheta \leftarrow 0$; ▷ no location selected yet
3: **while** $\vartheta < \alpha$ and $|M| > 0$ **do**
4: select $max(M_{i,j})$; ▷ location with more vehicles
5: $\Gamma \leftarrow \Gamma \cup M_{i,j}$; ▷ add location to solution
6: $\vartheta \leftarrow \vartheta + 1$; ▷ increase number of selected locations
7: remove $max(M_{i,j})$; ▷ remove selected location
8: **end while**
9: **return** Γ;

accounting for maximum number of vehicles. Notice that MCP-based does not use any mobility information. In order to improve the results of MCP-based, we propose modeling the deployment as a Probabilistic Maximum Coverage Problem.

MCP-based sorts the locations and selects the α most promising cells. Thus, the computational complexity of MCP-based is given by Equation 3.1.

$$Complexity(MCP - based) = \Theta(\alpha.M.\log\ M) \tag{3.1}$$

MCP-based is used as a baseline for our PMCP-based approach (detailed in the next section).

3.2 Probabilistic Maximum Coverage Problem

The Probabilistic Maximum Coverage Problem is a generalization of Maximum Coverage Problem: it assumes that each element (vehicle) has a probability p of belonging to a given set (location).

Definition 2 (Probabilistic Maximum Coverage Problem) *Suppose a collection of sets $S = \{S_1, S_2, ..., S_m\}$ defined over a domain of elements $X = \{x_1, x_2, ..., x_n\}$. Suppose the matrix P of size $m \times n$ where each $P_{m,n}$ gives the probability that element $x_n \in S_m$. Goal is to find a collection of sets $S' \subseteq S$ such that the expectancy of covered elements $E[|\bigcup_{S_i \in S'}|]$ is maximized.*

Suppose we have (let's say) 100 vehicles at location l_1. We observe that 20 vehicles migrate from l_1 to l_2. So, migration ratio from l_1 to l_2 is 0.2. Thus, $P(l_1, l_2) = 0.2$. P is a stochastic matrix responsible for modeling the mobility of vehicles (migration

ratios), and it represents the core of our proposed approach. Notice that tracking of vehicles is not necessary to infer P.

In this work we characterize the efficiency of using partial mobility information by comparing our approach to a MCP-based one. Since MCP-g relies on full knowledge of the vehicles trajectories, MCP-g is supposed to provide a better deployment of roadside units than our strategy. However, tracking individual vehicles seems not to be a realistic assumption in real deployments. Besides that, the selection of a deployment algorithm must take into account several other issues, such as computational cost: low-computational-cost strategies may support the dynamic deployment and real time deployment.

Although dynamic deployment is not possible with a stationary infrastructure, several works (Tonguz and Viriyasitavat, 2013; Jerbi et al., 2008; Luo et al., 2010; Annese et al., 2011; Mishra et al., 2011; Sommer et al., 2013) propose the adoption of a virtual-and-mobile infrastructure to support the vehicular communication. Going one step further, we may also consider the deployment as a metaphor to a wider class of resources allocation problems, and we easily find applications for real-time deployment, such as: (i) arrangement of resources for maximum efficiency in the network; (ii) management of content distribution (music, video, games); (iii) caching mechanisms (transferring contents in the right measure to vehicles during the trip), among several others.

Algorithm 3 presents our proposed algorithm.

M**:** Matrix describing the concentration of vehicles within the target region;

α**:** Number of available roadside units;

P**:** Migration ratios. Each $P_{i,j}$ stores the percentage of vehicles migrating from location i to location j.

PMCP-based is shown in Algorithm 3. Basically, it iteratively selects the locations presenting higher projections of the flow. When PMCP-based selects a location, it re-computes the number of distinct-and-uncovered vehicles at all precedent locations using the migration ratios (line 10). Line 3 starts projection (Λ) equal to the original flow. From lines 4-10 the heuristic loops selecting (at each iteration) the location offering the maximum expected number of contacts. The selected location is added to set Γ. The number of selected locations (ϑ) is incremented. After selecting a location to hold the new roadside unit, the heuristic (re) computes the flow projection into Λ.

Algorithm 3 PMCP-based

Input: M, α, P;
Output: Γ (urban cells receiving RSUs);
1: $\Gamma \leftarrow \emptyset$; ▷ solution-set starts empty
2: $\vartheta \leftarrow 0$; ▷ no places selected yet
3: $\forall i, j$ do $\Lambda_{i,j} = M_{i,j}$; ▷ projection starts equal to original flow
4: **while** $\vartheta < \alpha$ *and* $|M| > 0$ **do**
5: select $max(\Lambda_{i,j})$; ▷ largest expected # of contacts
6: $\Gamma \leftarrow \Gamma \cup M_{i,j}$; ▷ add location to solution
7: $\vartheta \leftarrow \vartheta + 1$; ▷ increase counter
8: remove $M_{i,j}$; ▷ remove location and compute projection
9: $\forall i, j$ do $\Lambda_{i,j} = GetProjectedFlow(i, j, M, P)$;
10: **end while**
11: **return** Γ;

The computational complexity of PMCP-based is given by Equation 3.2. The projection of the flow updates all the entries of Λ. For each entry we perform $\Theta(M)$ operations. Thus, we have $\Theta(M^2)$ Then, we sort Λ incurring in $\Theta(M.\log\ M)$. We repeat this processing α times.

$$Complexity(PMCP - based) = \Theta(\alpha.M^2) \tag{3.2}$$

PMCP-based differs from MCP-based with respect to the strategy used to compute the coverage, what explains the success of the strategy since that parameter defines where it is going to be deployed the next roadside unit. Next section discusses how the PMCP-based approach uses the probabilistic model to compute the volume of vehicles at each intersection.

3.2.1 Projection of the Flow

Projecting the flow has the purpose of discovering the number of vehicles reaching a given intersection. Basically, we count the number of vehicles at each one of the preceding intersections, and we obtain the projected number of vehicles using the information stored in the stochastic matrix P. We assume the stochastic matrix as the most reliable source of information, and we use P to reduce possible biases in the data by iterating the flow over time. In fact, the matrix P allow two complementary operations (pumping and pullout) (Silva et al., 2013, 2015a).

Pumping projection is when we combine the flow and the stochastic matrix aiming to achieve better estimates of traffic. Suppose that we have the estimate of vehicles at every intersection of a road network, but such estimate is not fully reliable.

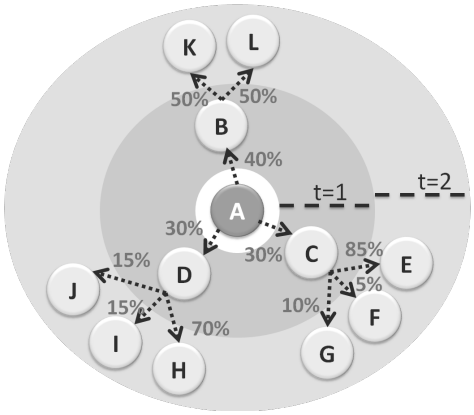
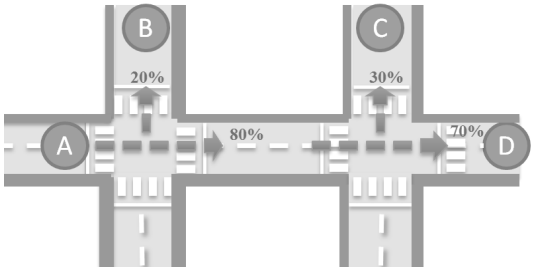


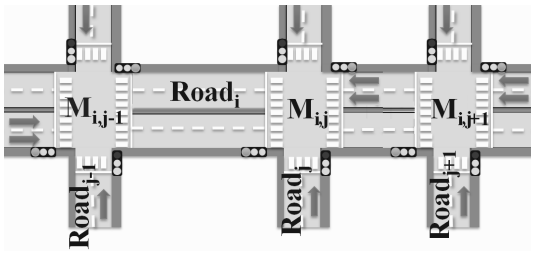
Figure 3.1. Probabilistic model. A generic vehicle in intersection A has 40% of probability to reach B, 30% to reach C and 30% to reach D when $t = 1$. If the vehicle chooses C, when $t = 2$ the vehicle has 85% of probability to reach E, 5% to reach F and 10% to reach G

Then, we may use the matrix P to fine tune the estimate of vehicles, pumping existing vehicles according to P during some time period.

Pullout projection is when we employ the matrix P to attenuate vehicles from the original estimate. Suppose that we have just inserted a new roadside unit within the road network. Then, we may use the pullout projection to remove all vehicles that will cross this newly deployed roadside unit: such vehicles will reach the infrastructure soon, thus, they can be pulled out from the model. Figure 3.1 illustrates the projection of flow: each circle indicates an intersection. Any vehicle within intersection A has 40% probability to reach B, 30% to reach C, and 30% to reach D after one unit of time (t=1). If the vehicle chooses C, then it has 85% probability to reach E, 5% to reach F, and 10% to reach G when t=2, and so forth.



(a) Hypothetical intersection.



(b) Two-way streets.

Figure 3.2. Projection of flow.

In this work we apply the pullout projection. Now we show another example illustrating the projection of flow. In Figure 3.2(a) we consider the existence of 100

vehicles at point A. Just to make this scenario as simple as possible, we arbitrarily consider vehicles moving in just two directions (left-to-right and bottom-up). The first intersection divides the flow: 20 vehicles to B, and 80 vehicles heading towards D. The next intersection divides the flow once again: 24 vehicles to C, and 56 to D. Our vision is that the snapshot of the initial moment in Figure 3.2(a) brings a distorted view of the flow, where A concentrates all the vehicles. But, when the flow is projected, vehicles are distributed along the road network, and we achieve a more precise view of the flow.

The projection of the flow has the purpose to better estimate the number of vehicles reaching a given intersection. In order to compute the projection $P_{i,j}$ for intersection $M_{i,j}$ we have to compute four different projections:

- Vehicles moving left-to-right;
- Right-to-left;
- Bottom-up;
- Top-down.

Let $P^u_{i,j}$ be a bi-dimensional matrix defined over $1..n, 1..n$ holding the likelihood of vehicles moving **up** at intersection $\{i, j\}$;

Let $P^d_{i,j}$ be a bi-dimensional matrix defined over $1..n, 1..n$ holding the likelihood of vehicles moving **down** at intersection $\{i, j\}$;

Let $P^r_{i,j}$ be a bi-dimensional matrix defined over $1..n, 1..n$ holding the likelihood of vehicles moving **right** at intersection $\{i, j\}$;

Let $P^l_{i,j}$ be a bi-dimensional matrix defined over $1..n, 1..n$ holding the likelihood of vehicles moving **left** at intersection $\{i, j\}$;

Let $P = (P^u, P^d, P^l, P^r)$ be a tridimensional stochastic matrix (migration ratios) holding P^u,P^d,P^l, and P^r.

$\forall i, j \in \{1, 2, .., n\}$ we have: $P^u_{i,j} + P^d_{i,j} + P^l_{i,j} + P^r_{i,j} = 1$

3.2.1.1 Left-to-Right Projection

The arrows on the left-side of Figure 3.2(b) represent vehicles moving left-to-right on road i towards intersection $M_{i,j}$. The number of vehicles ($LR^P_{i,j}$) arriving from

intersections $M_{i,j-1}$, $M_{i,j-2}$, ... , $M_{i,1}$ to $M_{i,j}$ is given by Eq. 3.3.

$$LR^P_{i,j} = \sum_{k=0}^{j-1}(UncoveredVehicles(M_{i,k}).\prod_{l=k}^{j-1} P^r_{i,l}) \tag{3.3}$$

$\{i, j\}$ represents the intersection of vertical road i and horizontal road j.

$UncoveredVehicles()$ returns the expected number of vehicles not covered crossing a given intersection;

Product computes the compound probability resulting from a series of intersections.

3.2.1.2 Right-to-Left Projection

The arrows on the right side of Figure 3.2(b) represent vehicles moving right-to-left on road i towards intersection $M_{i,j}$. The number of vehicles ($RL^P_{i,j}$) arriving from intersections $M_{i,j+1}$, $M_{i,j+2}$, ... , to $M_{i,j}$ is given by Eq. 3.4.

$$RL^P_{i,j} = \sum^{\forall k>j}(UncoveredVehicles(M_{i,k}).\prod_{l=j+1}^{k} P^l_{i,l}) \tag{3.4}$$

3.2.1.3 Bottom-Up Projection

The bottom-up arrows in Figure 3.2(b) represent vehicles moving bottom-up on road j. The number of vehicles ($BU^P_{i,j}$) arriving from intersections $M_{i-1,j}$, $M_{i-2,j}$, ... , $M_{1,j}$ to $M_{i,j}$ is given by Eq. 3.5.

$$BU^P_{i,j} = \sum_{k=0}^{i-1}(UncoveredVehicles(M_{k,j}).\prod_{l=k}^{i-1} P^u_{j,k}) \tag{3.5}$$

3.2.1.4 Top-Down Projection

The top-down arrows in Figure 3.2(b) represent vehicles moving top-down on road j. The number of vehicles ($TD^P_{i,j}$) arriving from intersections $M_{i+1,j}$, $M_{i+2,j}$, ... , to $M_{i,j}$ is given by Eq. 3.6.

$$TD^P_{i,j} = \sum^{\forall k>i}(UncoveredVehicles(M_{k,j}).\prod_{l=i+1}^{k} P^d_{j,k}) \tag{3.6}$$

Therefore, the total projected flow ($PF^P_{i,j}$) may be estimated by adding $LR^P_{i,j}$, $RL^P_{i,j}$, $BU^P_{i,j}$ and $TD^P_{i,j}$ to the number of uncovered vehicles found in intersection $M_{i,j}$, as Eq. 3.7 shows.

$$PF^P_{i,j} = UncoveredVehicles(M_{i,j}) + LR^P_{i,j} + RL^P_{i,j} + BU^P_{i,j} + TD^P_{i,j} \tag{3.7}$$

Equation 3.7 computes number of uncovered vehicles reaching the intersection $\{i, j\}$ from all four possible directions. When computing the projection of the flow, the influence of distant vehicles decays fast: distant vehicles are less likely to reach the selected intersection (since they may choose among several possible paths). On the other hand, the number of distant vehicles increases very fast as we increase the distance from the selected intersection. The relation between these magnitudes determines the expected number of contacts, defining the intersection that will receive the next roadside unit. Finally, such choice defines the efficiency of the entire method. Next section derives our theoretical grid-based road network.

3.3 Analytic Model for the Theoretical Grid Road Network

In this section we derive an analytic model of a theoretical grid-based road network aiming to obtain M, i.e., the expected number of vehicles at every position of the grid. Our goal is to derive the number of expected vehicles at every position of the grid considering:

(a) Incoming flows;

(b) Stochastic matrix P, i.e., turning ratios.

3.3.1 Model #1: One Way Roads and Absence of Sinks

Here we derive an analytic model for a theoretical grid-based road network composed of one-way roads where vehicles always move top-down and left-right in the grid. We also assume the non-existence of sinks.

Let R be a perfect grid defined over $1..n \times 1..n$ composed of:

- n one-way vertical roads numbered from 1 to n (v_1, v_2, ..., v_n);

- n one-way horizontal roads numbered from 1 to n (h_1, h_2, ..., h_n);

Figure 3.3 present R (road network).

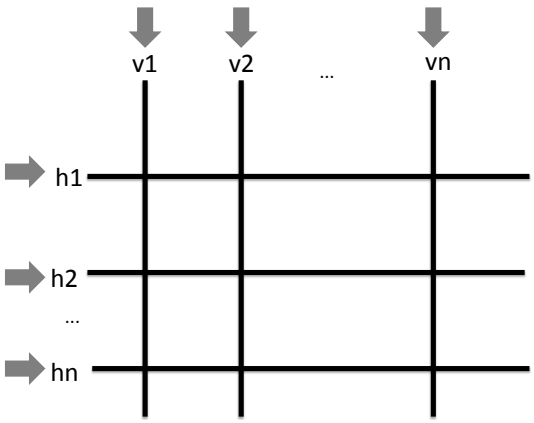


Figure 3.3. Theoretical road network containing one way roads: vehicles move top-down and left-right.

Let $V = \{v_1, v_2, ..., v_n\}$ be the set of vertical roads.
Let $H = \{h_1, h_2, ..., h_n\}$ be the set of horizontal roads.
Thus, $R = \{v_1, v_2, ..., v_n\} \cup \{h_1, h_2, ..., h_n\}$.

Let's also assume:

- Each road $r \in R$ has an independent level of traffic;
- R has no sinks: every vehicle entering R also leaves R after a long enough time interval. In practical terms, we are assuming that vehicles never park;
- We arbitrarily define vehicles moving top-down and left-right within the grid: all roads are one-way for definition;
- Vehicles enter R according to Poisson Process, a valid assumption, also used by several works in the literature such as (Chun et al., 2010; Larson, 1982; McNeil, 1968; Gerlough and Schuhl, 1955). Thus:
  - Level of traffic in vertical roads is $\lambda_{v,k}$, where $k \in \{1, 2, ..., n\}$;
  - Level of traffic in horizontal roads is $\lambda_{h,k}$, where $k \in \{1, 2, ..., n\}$;

Figure 3.4 indicates the level of traffic at grid borders.

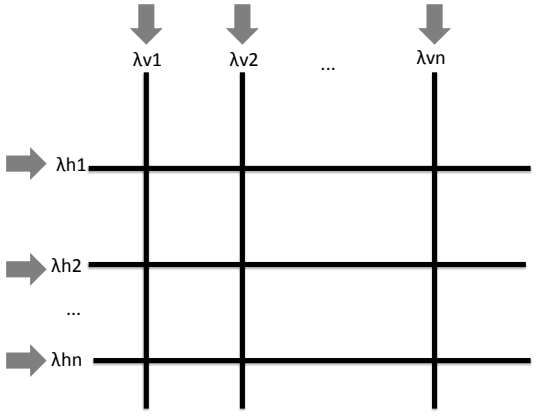


Figure 3.4. Theoretical road network: Each road has a different level of traffic modeled as a Poisson Process.

Recall that a vehicle has just two possible moves: (a) vertical; and (b) horizontal.
Let P be a bi-dimensional stochastic matrix defined over $1..n \times 1..n$.
$P_{i,j}$ holds the likelihood of vertical movements at intersection $\{i, j\}$.
$1 - P_{i,j}$ is the likelihood of horizontal movements at $\{i, j\}$.
Figure 3.5 indicates matrix P.

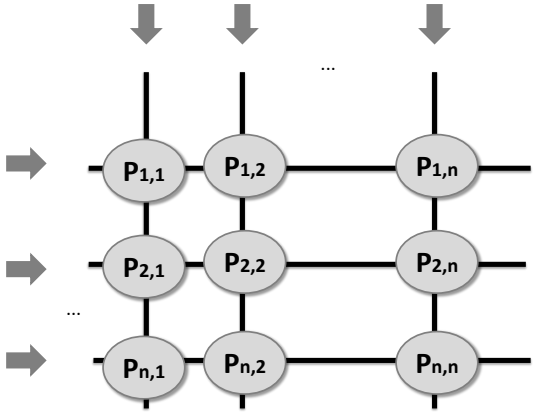


Figure 3.5. Theoretical road network: Matrix P holding turning ratios.

Let's extend P to be defined at grid borders.
P becomes defined over $0..n \times 0..n$.
Let $P_{0,k} = 1$ and $P_{k,0} = 0 \; \forall k \in \{1, 2, .., n\}$.

Figure 3.6 indicates extended matrix P.

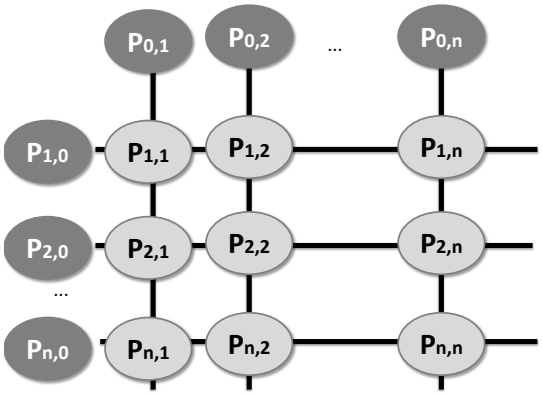


Figure 3.6. Theoretical road network: Extended matrix P.

Let M be a bi-dimensional matrix defined over $1..n, 1..n$.
M holds the **expected** number of vehicles at each intersection of R.
Our goal is to derive an analytic expression for M.

Figure 3.7 indicates matrix M.

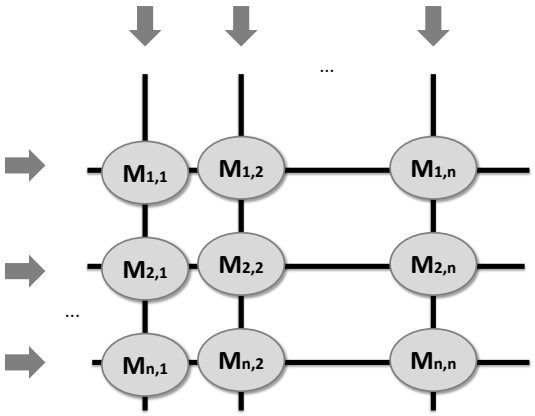


Figure 3.7. Theoretical road network: Matrix M.

Let's extend M to contemplate grid borders.
M becomes defined over $0..n \times 0..n$.
Let $M_{0,k} = \lambda_{v,k}\ \forall k \in \{1, 2, .., n\}$.
Let also $M_{k,0} = \lambda_{h,k}\ \forall k \in \{1, 2, .., n\}$.

Figure 3.8 indicates matrix M.

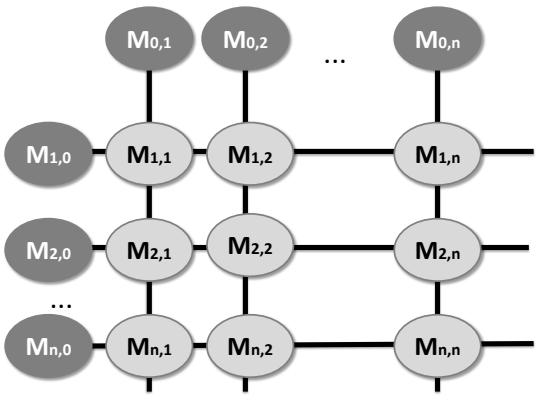


Figure 3.8. Theoretical road network: Extended matrix M.

According to Larson (1982), if P(λ_1) and P(λ_2) are Poisson Processes, then:

$$P(\lambda_1) + P(\lambda_2) = P(\lambda_1 + \lambda_2) \tag{3.8}$$

Larson (1982) also states if X and Y are independent random variables, then:

$$E(XY) = E(X).E(Y) \tag{3.9}$$

Thus, because the number of vehicles reaching a given intersection is independent from the turning likelihood, M may be recursively defined using Equation 3.10:

$$M_{a,b} = M_{a-1,b}.P_{a-1,b} + M_{a,b-1}.(1 - P_{a,b-1}) \tag{3.10}$$

Where:

$M_{0,k} = \lambda_{v,k}\ \forall k \in \{1, 2, ..., n\}$;
$M_{k,0} = \lambda_{h,k}\ \forall k \in \{1, 2, ..., n\}$;
$P_{0,k} = 1\ \forall k \in \{1, 2, ..., n\}$;
$P_{k,0} = 0\ \forall k \in \{1, 2, ..., n\}$;
$M_{i,j}$ holds the number of vehicles in position $\{i, j\}$ of the grid;
$P_{i,j}$ holds the likelihood of vertical movements at intersection $\{i, j\}$;
$(1 - P_{i,j})$ holds the likelihood of horizontal movements at intersection $\{i, j\}$;
Every road $r \in R$ is one-way;
Vehicles move left-to-right or top-down at each intersection of R;
Sources of vehicles are top and left borders of R;
There are no sink of vehicles in R.

Suppose new variables P^v and P^h also defined over $0..n, 0..n$, where:
$P^v_{a,b} = P_{a,b}$, $\forall a, b \in \{1, 2, ..., n\}$;
$P^h_{a,b} = 1 - P_{a,b}$, $\forall a, b \in \{1, 2, ..., n\}$;

Then, as an immediate result:

$$P^v_{a,b} + P^h_{a,b} = 1, \forall a, b \in \{1, 2, ..., n\} \tag{3.11}$$

Thus, Equation 3.10 may be written as:

$$M_{a,b} = M_{a-1,b}.P^v_{a-1,b} + M_{a,b-1}.P^h_{a,b-1} \tag{3.12}$$

Where:

$P^v i, j$ holds the likelihood of vertical movements at intersection $\{i, j\}$;
$P^v_{0,k} = 1$ $\forall k \in \{1, 2, ..., n\}$;
$P^v_{k,0} = 0$ $\forall k \in \{1, 2, ..., n\}$;
$P^h_{i,j}$ holds the likelihood of horizontal movements at intersection $\{i, j\}$.
$P^h_{k,0} = 1$ $\forall k \in \{1, 2, ..., n\}$;
$P^h_{0,k} = 0$ $\forall k \in \{1, 2, ..., n\}$;
$P^v_{a,b} + PH_{a,b} = 1$, $\forall a, b \in \{1, 2, ..., n\}$.
$M_{0,k} = \lambda_{v,k}$ $\forall k \in \{1, 2, ..., n\}$;
$M_{k,0} = \lambda_{h,k}$ $\forall k \in \{1, 2, ..., n\}$;
$M_{i,j}$ holds the number of vehicles in position $\{i, j\}$ of the grid;

3.3.2 Model #2: One Way Roads with Sinks

Now we extend our model to also contemplate sinks:

Let P^s be a bi-dimensional stochastic matrix defined over $1..n \times 1..n$.

$P^s_{i,j}$ holds the likelihood of vehicles being dragged by a sink at intersection $\{i, j\}$.

Considering P^s, Equation 3.11 is replaced by Equation 3.13:

$$P^v_{a,b} + P^h_{a,b} + P^s_{a,b} = 1, \forall a, b \in \{1, 2, ..., n\} \tag{3.13}$$

Equation 3.13 states that a vehicle has three possible choices: (a) move in vertical (top-down); (b) move in horizontal (left-right); (c) park forever.

Thus, model #2 is:

$$M_{a,b} = M_{a-1,b}.P^{v}_{a-1,b} + M_{a,b-1}.P^{h}_{a,b-1} \tag{3.14}$$

Where:

$P^{s}_{i,j}$ holds the likelihood of vehicles being dragged by a sink at intersection $\{i, j\}$.
$P^{v}i, j$ holds the likelihood of vertical movements at intersection $\{i, j\}$;
$P^{v}_{0,k} = 1\ \forall k \in \{1, 2, ..., n\}$;
$P^{v}_{k,0} = 0\ \forall k \in \{1, 2, ..., n\}$;
$P^{h}_{i,j}$ holds the likelihood of horizontal movements at intersection $\{i, j\}$.
$P^{h}_{k,0} = 1\ \forall k \in \{1, 2, ..., n\}$;
$P^{h}_{0,k} = 0\ \forall k \in \{1, 2, ..., n\}$;
$P^{v}_{a,b} + P^{h}_{a,b} + P^{s}_{a,b} = 1,\ \forall a, b \in \{1, 2, ..., n\}$.
$M_{0,k} = \lambda_{v,k}\ \forall k \in \{1, 2, ..., n\}$;
$M_{k,0} = \lambda_{h,k}\ \forall k \in \{1, 2, ..., n\}$;
$M_{i,j}$ holds the number of vehicles in position $\{i, j\}$ of the grid;

3.4 Analytic Formulation of Infrastructure Deployment

Considering the previous discussion, now we formulate the allocation of roadside units aiming to maximize the number of distinct vehicles contacting the infrastructure in a grid-based road network as follows:

Let $R = V \cup H$ be a grid-based road network composed of n vertical roads (V) and n horizontal roads (H). Let $A = \{a_1, a_2, ..., a_h\}$ represent the set of h vehicles. Let M be a bi-dimensional matrix defined over $1..n, 1..n$ representing the number of vehicles at each intersection (v, h), $\forall v \in V$ and $\forall h \in H$. Available roadside units is given by α. $\Gamma \leftarrow \emptyset$ is our solution set, i.e., location of each roadside unit. Let $C \leftarrow \emptyset$ represents vehicles 'covered'. $\forall a \in A \quad C \leftarrow C \cup a$ if a has crossed at least one of the intersections composing Γ.

Our goal is to find out $\Gamma \subseteq M$ where $|\Gamma| \leq \alpha$ and $|C|$ is maximum.

Algorithm MCP-based proposes the following solution:

Let $S \leftarrow sort(M)$

$\Gamma \leftarrow Max(S, \alpha)$

MCP-based selects the top-α intersections presenting higher traffic.

Algorithm PMCP-based proposes the following solution:

Let $P^u_{i,j}$ be a bi-dimensional matrix defined over $1..n, 1..n$ holding the likelihood of vehicles moving **up** at intersection $\{i, j\}$;

Let $P^d_{i,j}$ be a bi-dimensional matrix defined over $1..n, 1..n$ holding the likelihood of vehicles moving **down** at intersection $\{i, j\}$;

Let $P^r_{i,j}$ be a bi-dimensional matrix defined over $1..n, 1..n$ holding the likelihood of vehicles moving **right** at intersection $\{i, j\}$;

Let $P^l_{i,j}$ be a bi-dimensional matrix defined over $1..n, 1..n$ holding the likelihood of vehicles moving **left** at intersection $\{i, j\}$;

Let $P = (P^u, P^d, P^l, P^r)$ be a tridimensional stochastic matrix (migration ratios) holding P^u,P^d,P^l, and P^r.

$\Gamma \leftarrow \emptyset$ is our solution set, i.e., location of each roadside unit.

Let $C^{P,k}_{M,\Gamma}$ represent $PF^P_{m,\gamma}$ $\forall m \in M$, where γ is the k^{th} element of Γ.

Where:

$PF^P_{i,j} = UncoveredVehicles(M_{i,j}) + LR^P_{i,j} + RL^P_{i,j} + BU^P_{i,j} + TD^P_{i,j}$

$LR^P_{i,j} = \sum_{k=0}^{j-1}(UncoveredVehicles(M_{i,k}).\prod_{l=k}^{j-1} P^r_{i,l})$

$RL^P_{i,j} = \sum^{\forall k>j}(UncoveredVehicles(M_{i,k}).\prod_{l=j+1}^{k} P^l_{i,l})$

$BU^P_{i,j} = \sum_{k=0}^{i-1}(UncoveredVehicles(M_{k,j}).\prod_{l=k}^{i-1} P^u_{j,k})$

$TD^P_{i,j} = \sum^{\forall k>i}(UncoveredVehicles(M_{k,j}).\prod_{l=i+1}^{k} P^d_{j,k})$

Our goal is to iteratively transfer element $\{i, j\}$ from M to Γ where $PF^P_{i,j}$ is maximum for all $(i, j) \in M$ until $|\Gamma| = \alpha$.

3.5 Simulations and Results

Now we present an initial comparison of the PMCP-based and MCP-based approaches considering a grid-based road network. Our goal is to demonstrate a didactic explanation of our deployment algorithm. Here we assume that roads intersections

define "locations", and we compute the projection of the flow at intersections. Thus, **in this set of experiments we consider stochastic matrix P holding turning ratios at roads intersections**. Our road network is composed of 9x9 roads, as illustrated in Figure 5.3.

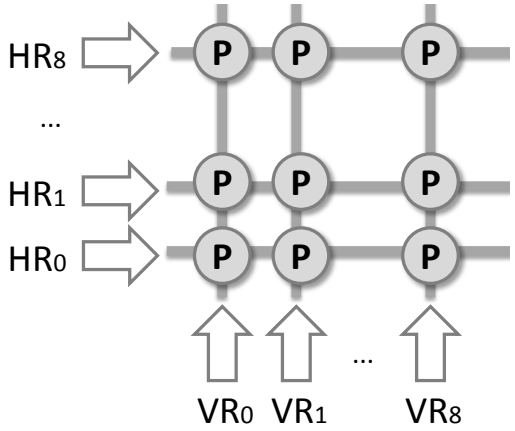


Figure 3.9. 9x9 Theoretical Grid-like Road Network.

We built a simulator emulating such road network. Simulator inserts vehicles at bottom and left borders of the grid. When vehicles reach top/right borders, they leave the simulation. Our simulator dispatches one vehicle at time in a random walk trip inside the grid. The random walk is guided by the stochastic matrix P (turning ratios at roads intersections). The simulator works in the following manner:

In **step 1** we get a random number to define where the vehicle will start the trip. There are 18 possible start points: nine start points in bottom borders ($VR_0..VR_8$), and nine start points in left borders ($HR_0..HR_8$).

In **step 2**, vehicle is into the grid and trip starts. While the vehicle is still inside the grid, we loop: we get a random number. Using this random number we decide if the vehicle has to move up or right. Vehicle always moves between adjacent cells. Every time a vehicle gets inside a cell, we increment the 'counter of visits' for that cell. When vehicle leaves the grid, we return to **step 1**.

We repeat this process β times, where β represents the number of vehicles that we intend to simulate. Then, we apply the PMCP-based and MCP-based approaches. We investigate the benefits of incorporating the probabilistic approach through five experiments, now using just the MCP-based approach as our baseline.

3.5.1 Trips

Experiment considers 100 random scenarios in a grid-based road network. We vary the capacity (λ) in order to generate several road networks. We plot the mean and the standard deviation for the number of contacts between vehicles and the infrastructure. Figure 3.10 shows the number of contacts for the PMCP-based and MCP-based approaches: y-axis presents the number of contacts between vehicles and the infrastructure, while the x-axis shows the number of the deployed roadside units. We indicate the number of contacts as a percentage of all trips launched (because the number of contacts depends on the duration of the simulation). Result shows that when we deploy roadside units in approximately 6.2% of intersections, PMCP-based achieves one contact per trip: such result does not imply that every vehicle reaches exactly one roadside unit. In fact, some vehicles never reach roadside units, while others experience several contacts (see Experiment #6). The selection of the intersections made by the PMCP-based approach offers more vehicle-to-infrastructure contacts.

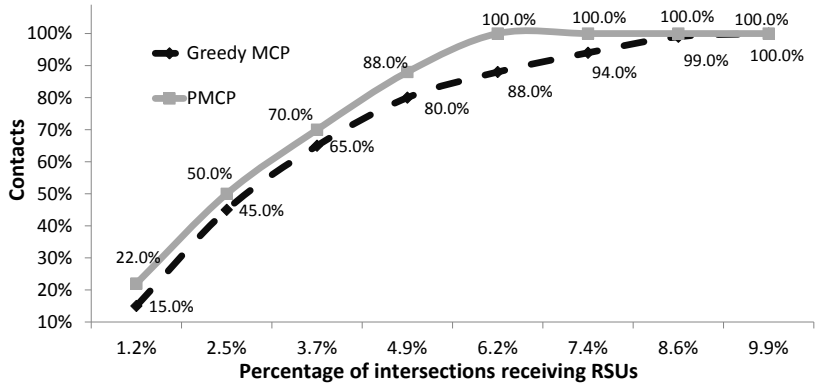


Figure 3.10. Number of vehicle-to-infrastructure contacts: x-axis indicates number of intersections receiving roadside units (indicated as a percentage of intersections). The y-axis indicates number of contacts (indicated as a percentage of total trips). The graph is truncated in 100%.

Figure 3.11 indicates the standard deviation (always $\leq 2.1\%$).

3.5.2 Allocation of Roadside Units

This experiment reveals how both heuristics allocate the roadside units within the region. We do this by: (i) picking one scenario at random; (ii) running the simulation; and, (iii) analyzing how each technique places each roadside unit. We assume the flow of vehicles shown in Figure 3.12.

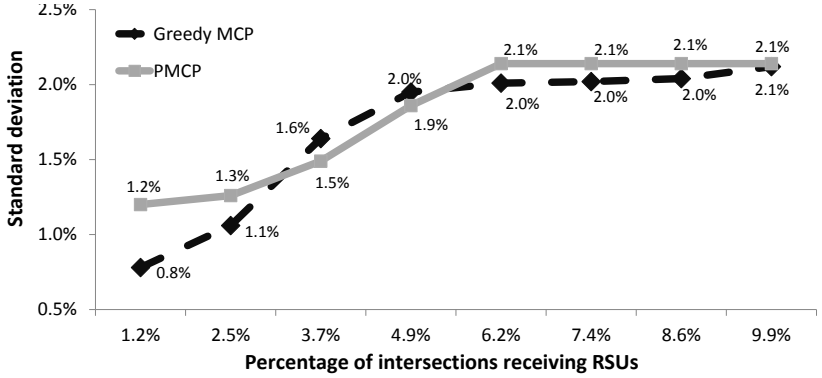


Figure 3.11. Standard deviation.

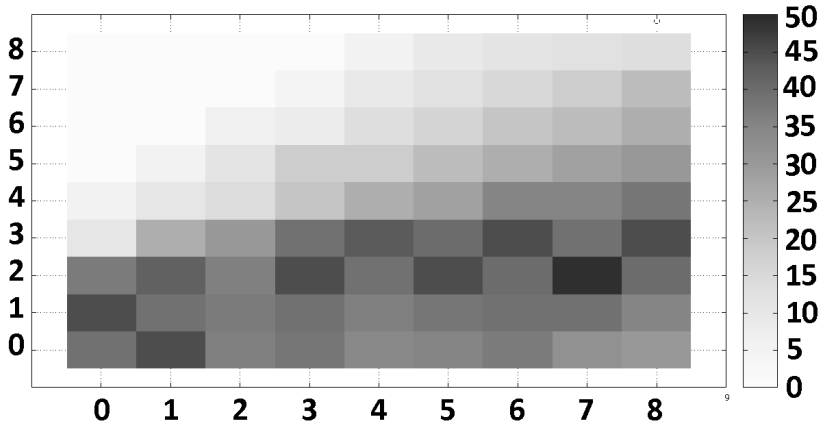


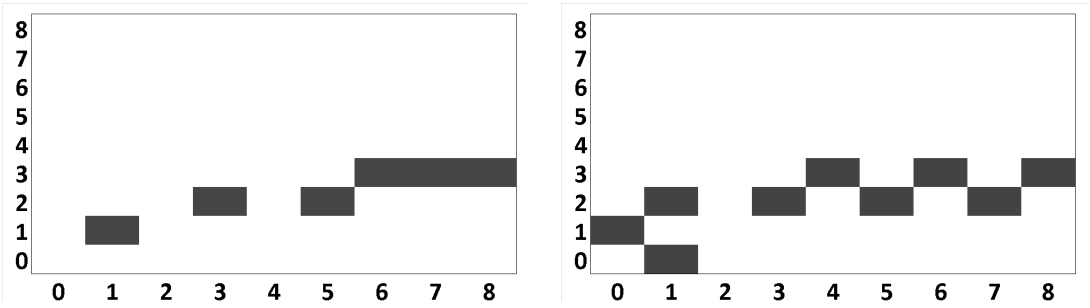
Figure 3.12. Random Scenario: We use a 9×9-grid and random stochastic matrix.

PMCP-based allocates roadside units according to layout shown in Figure 3.13(a): layout offers the average of one contact per trip. The layout of the MCP-based approach is shown in Figure 3.13(b) with 0.892 contacts per trip.

3.5.3 Coverage Map

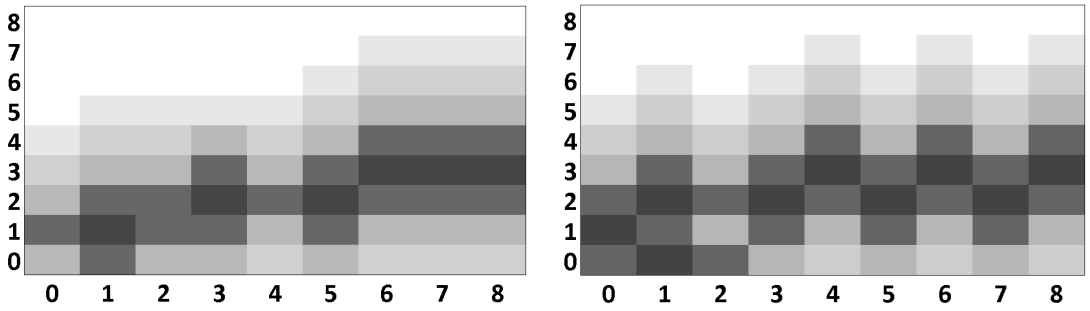
Figure 3.14 shows the coverage map for both algorithms. We use heatmaps to represent the road network, and gray scale to indicate the distance from each intersection to the nearest roadside unit. Figure 3.14 characterizes the road network in terms of the distance to the nearest roadside unit. The scale has five levels, starting from black ($distance = 0$) to white ($distance > 4$).

When we contrast Figures 3.12 and 3.14 we notice that the coverage map of the PMCP-based approach fits better the traffic pattern (especially when we consider that PMCP-based deploys just six roadside units while MCP-based deploys nine roadside



(a) PMCP-based (b) MCP-based

Figure 3.13. Location of RSUs according to both algorithms.



(a) PMCP-based (b) MCP-based

Figure 3.14. Coverage map based on distance: map characterizes all intersections in terms of distance to nearest roadside unit in a 5-level scale: from black ($distance = 0$) to white ($distance > 4$).

units). Since MCP-based is concerned only with local the density of vehicles, it may allocate roadside units in overlaid areas. As an example, MCP-based places roadside units at coordinates $\{1, 0\}$ and $\{1, 2\}$. PMCP-based replaces these for a single roadside unit located at $\{1, 1\}$.

3.5.4 Vehicle-to-Infrastructure Distance

Now we provide a measure of the vehicle-to-infrastructure distance, yet considering the same scenario of Figure 3.12. Vehicles dist=0 are positioned at an intersection containing a roadside unit. Vehicles dist=1 are 1-intersection away from the nearest roadside unit, and so forth. Figure 3.15 indicates that the distance from vehicles to the nearest roadside unit: the x-axis indicates distance, while the y-axis indicates the percentage of vehicles.

PMCP-based has 41% of the vehicles $dist = 1$, while MCP-based has only

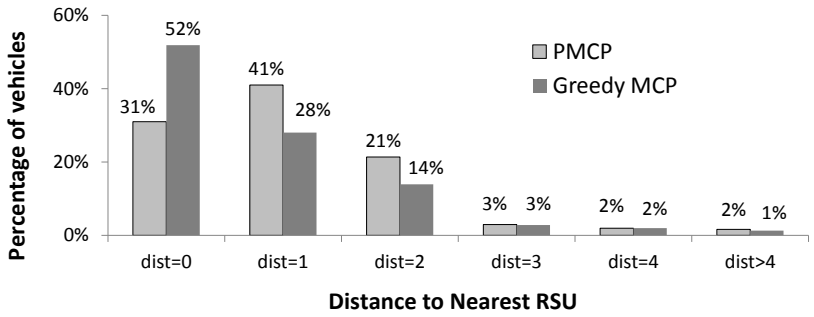


Figure 3.15. Histogram Vehicle-to-RSUs Distance: characterizes the position of each vehicle according to the nearest RSU.

28%. The same behavior repeats for vehicles $dist = 2$ (PMCP-based=21% $\times$ MCP-based=14%). These results are a consequence of the projection of the flow proposed by PMCP-based: the position of each roadside unit is defined in order to achieve the largest number of vehicles.

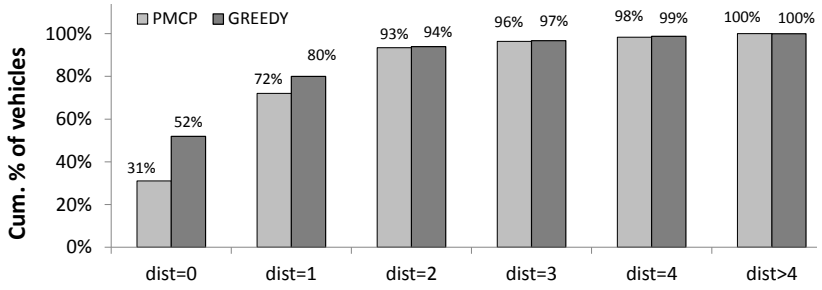


Figure 3.16. Cumulative percentage of vehicles x Distance to nearest roadside unit.

Figure 3.16 shows the cumulative perspective of previous result: the x-axis presents the cumulative distance to roadside units, while the y-axis indicates the cumulative percentage of vehicles allocated that far. When considering distance ≥ 1, PMCP-based and MCP-based show similar performances.

3.6 Analysis

So far we have conceptualized the Probabilistic Maximum Coverage Problem and we have proposed a constructive heuristic that approximates the solution to PMCP assuming that we do not have access to vehicles trajectories (PMCP-based). We compare PMCP-based to MCP-based: MCP-based is an approximate solution to Maximum Coverage Problem. The only step it does is to sort the intersections and select those intersections presenting highest number of vehicles. Our approximated

solution to PMCP-based initially selects the most crowded location (in this scenario, an intersection). Then, it iteratively selects the location presenting the highest project traffic. Every time our solution selects an intersection, we remove portions of vehicles that will drive through this recently added roadside unit, and then we re-compute the projection for all remaining intersections. The most important assumption of our work is we do not have access to vehicles trajectories. We have presented a comparison between PMCP-based and MCP-based considering an unrealistic road network composed of a grid of just 9×9 roads. Although such road network is very distant from reality, these experiments demonstrate the basic principles that will guide our investigation in the following sections. Result show that partial mobility information really increases the deployment quality in such unrealistic road network.

- We demonstrate that the traffic flow is directly related to stochastic matrix P. Such issue motivates us to study the gains achieved when using partial mobility information, i.e., the global behavior of vehicles without tracking individual vehicles;
- We demonstrate that PMCP-based requires just six roadside units to achieve one contact per trip, while MCP-based requires nine roadside units to achieve 0.892 contacts per trip;
- We demonstrate that PMCP-based provides a coverage map better fitting vehicular mobility;
- We demonstrate that PMCP-based provides more contact opportunities for vehicles during a typical trip. We also demonstrate that coverage provided by PMCP-based has a deviation one order of magnitude lower than MCP-based.

3.7 Overview of the Next Chapter

In the next chapter we propose partitioning the road network: we reduce any road network into a grid-based road network in order to represent complex road networks.

Chapter 4

Real Road Network and Random Flow Deployment

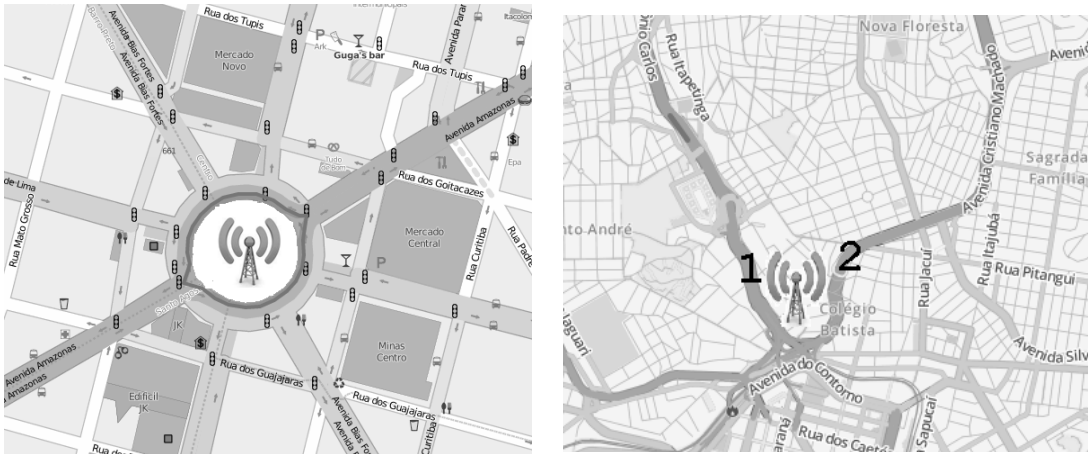
In Chapter 3 we have demonstrated that a PMCP-based approach increases the deployment performance when compared to a MCP-based one. As our PMCP-based approach counts with the projection of the flow, it allocates the roadside units in accordance with the flow of vehicles, and optimizes the number of required roadside units in order to achieve a given vehicles share. Until now we have evaluated the PMCP-based approach just over a theoretical grid-based road network. Our initial formulation of the deployment problem has one important limitation: the roadside units are always deployed at roads intersections. Although that seems reasonable for several scenarios, we cannot assume that such a premise is valid for every possible road network. For instance, some road network may be better served if we place a roadside unit in the center of an important square to cover vehicles surrounding it, such as illustrated in Figure 4.1(a). Or, perhaps, a roadside unit may be used to serve two important avenues, such as presented in Figure 4.1(b).

In this chapter we present the partition technique as our proposal to handle arbitrary road networks. We discrete the continuous road network using the same principles guiding the digitalization of pictures. The goal of this chapter is:

1. Proposal of a strategy able to represent arbitrary road networks

2. Present a worst case analysis of the deployment efficiency.

In this work we present a strategy for the deployment of roadside units considering the partition of the road network into same size urban cells. After partitioning the road network we propose a deployment algorithm that considers partial mobility



(a) Roadside unit in a square. (b) One roadside unit serving two avenues.

Figure 4.1. Placing RSUs not in intersections. Maps from `http://openstreetmap.org`.

information. In order to present a worst case analysis, we consider a real road network and a synthetically generated traffic where destinations of vehicles are randomly chosen. Our main findings are:

- By covering 5% of the road network we achieve 90% of all trips experiencing at least one contact opportunity with the infrastructure;
- There is an upper bound in the number of partitions for a given region: over-partitioning does not improve the deployment accuracy indefinitely;
- Partial mobility information improves fairness in vehicle-to-infrastructure (V2I) contacts;
- As traffic changes over time, we may rely on dynamic methods to improve the allocation of non-stationary roadside units: if we recalculate the position of each roadside unit every 1,000s, we improve the vehicle-to-infrastructure contact time in almost 60%.

The chapter is organized as follows: Section 4.1 discusses the partition strategy. Section 4.2 presents the algorithms for partitioning a road network. Section 4.3 adapts the projection of the flow to consider partitions. Section 4.4 presents the methodology of evaluation. Sections 4.5 and 4.6 details our experiments. Section 4.7 presents our remarks, and Section 4.8 presents an overview of the next chapter.

4.1 Partitioning Road Networks

The partition strategy divides the urban area in a set of adjacent same-size cells. Once the road network is partitioned, we discard the (complex) road network, and we manipulate the flow between adjacent grid cells in a divide-and-conquer strategy. Then, we apply the PMCP-based and MCP-based approaches in order to select the most promising partitions, and we compare the performance of each approach through several metrics discussed from Sections 4.4 to 4.6. Some studies have already proposed partitioning the urban area. Typically, these efforts use partitioning as an intermediate step to accomplish:

(a) Simplified analytic models (Aslam et al., 2012);

(b) Subdivision of the problem into several smaller ones (Habib and Safar, 2007);

(c) Simplified road network consisting of only horizontal and vertical roads (Xie et al., 2013).

The literature presents several deployment algorithms for roadside units. But, it is not unusual that these algorithms do not receive the road network as input. Such issue happens because a road network is a very complex structure, and handling such complex structure is not so simple. Nevertheless, the road network is a key information when we analyze the traffic flow. Differently from other approaches (Aslam et al., 2012; Xie et al., 2013), we perceive the grid as a smart, intuitive and economic strategy to represent any road network. The allocation of roadside units to support the vehicular communication is a NP-Hard problem: the exact solution requires checking all the possible combinations of roadside units at all possible locations of the urban area in order to select those locations maximizing some objective function.

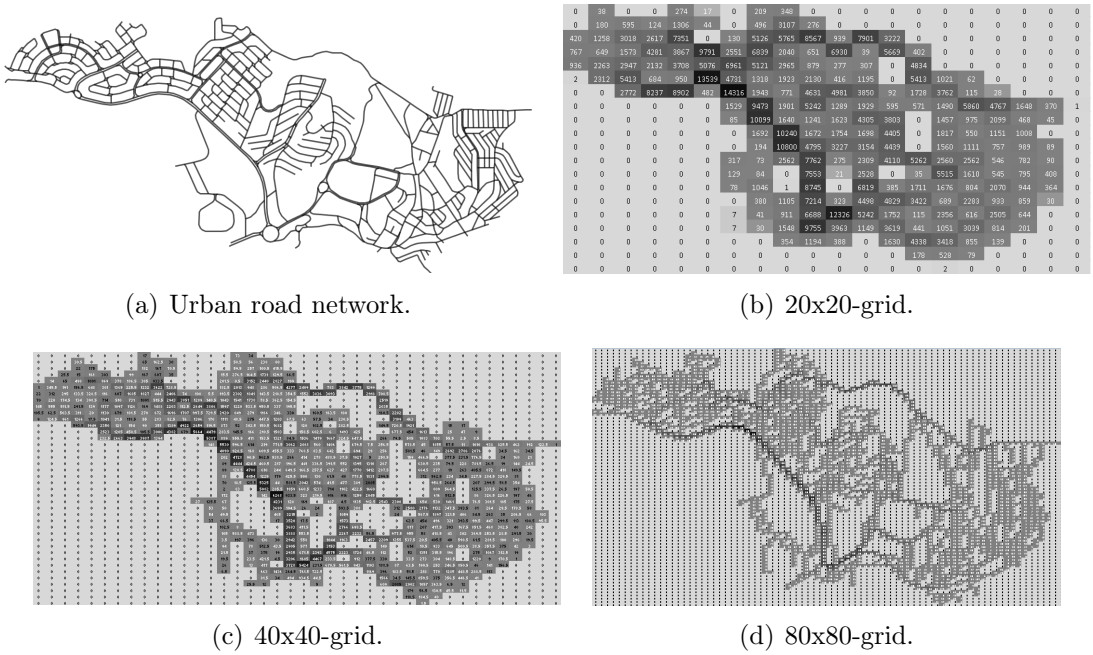
$$C_{\alpha}^{|M|} = \frac{|M|!}{\alpha!(|M| - \alpha)!} \tag{4.1}$$

Where:

M: number of possible locations to receive roadside units;

α: number of roadside units for deployment.

The parameters to be maximized depend upon the purposes of the vehicular network. Among a huge number of possibilities, network designers may, for instance, maximize the connected time of vehicles traveling a given route, or create a single contiguous area where the possibilities of disconnections are negligible. Network



(a) Urban road network. (b) 20x20-grid.

(c) 40x40-grid. (d) 80x80-grid.

Figure 4.2. Plot shows the improvement in accuracy of the model when we increase the grid setup.

designers may also maximize the number of times that each vehicle gets in contact with the infrastructure, or maximize the number of distinct vehicles that reach any roadside unit (an interesting metric when we intend to collect and disseminate traffic announcements in large scale).

Figure 4.2(a) shows the map of a real city[1] with approximately 10km^2. Figure 4.2(b) shows that such road network may be overlapped with a grid presenting cells of arbitrary size. In this specific scenario, we have overlapped a 20x20-grid (400 urban cells). Inside each urban cell we indicate the number of vehicles crossing each urban cell during a certain period of time. Once the road network is partitioned, we discard the original road network, and we manipulate the flow between adjacent cells. Now each urban cell is mapped to a grid cell where vehicles (always) move between adjacent cells. And, as we do not track individual vehicles, the only information we have is the number of vehicles migrating from/to adjacent cells. Figure 4.2(c) overlaps a 40x40-grid, while Figure 4.2(d) overlaps an 80x80-grid.

[1]Ouro Branco, Minas Gerais State, Brazil.

Figure 4.2 indicates a rectangular shape for urban cells, a little bit unusual shape[2]. In fact, our goal is not to represent ranges of 'coverage', but to divide-and-conquer the road network. We split the road network into urban cells. After that, we select those urban cells resulting in better coverage. The exact location of each roadside unit within a given urban cell may be discovered by recursive applications of our strategy, until reaching the desired level of granularity. However, a practical intra-cell deployment must also take into account another issues, such as:

(a) Availability of energy supply;

(b) Signal interference due to existent wireless networks;

(c) Large constructions blocking signals;

As we reduce the dimensions of urban cells, we increase the resolution of the grid, making it increasingly similar to the original road network.

4.2 Algorithmic Description

Partitioning requires the following steps:

(a) Partition the road network;

(b) Translate mobility trace from real coordinates to grid coordinates;

(c) Compute density of vehicles in each urban cell;

(d) Compute stochastic matrix (migration ratios) P.

Algorithm 4 presents the strategy to partition urban roads. Below we present input parameters:

ϱ**:** Road network (geographical map);

ψ_x**:** Number of horizontal urban cells;

ψ_y**:** Number of vertical urban cells.

[2]Note that the coverage area of an infrastructure node cannot be modeled as a circular shape because the intensity may not reflect the quality of the experienced connectivity (Cataldi and Harri, 2011).

Algorithm 4 Partition the Road Network.

Input: ϱ, ψ_x, ψ_y;
Output: Δ (grid representing road network);
1: $min\_\phi_x \leftarrow min\_x\_coordinate(\varrho)$
2: $min\_\phi_y \leftarrow min\_y\_coordinate(\varrho)$
3: $max\_\phi_x \leftarrow max\_x\_coordinate(\varrho)$
4: $max\_\phi_y \leftarrow max\_y\_coordinate(\varrho)$
5: $size\_\phi_x \leftarrow (max\_\phi_x - min\_\phi_x)/\psi_x$
6: $size\_\phi_y \leftarrow (max\_\phi_y - min\_\phi_y)/\psi_y$
7: $\Delta \leftarrow \{\{min\_\phi_x, min\_\phi_y\}, \{max\_\phi_x, max\_\phi_y\}, \{size\_\phi_x, size\_\phi_y\}\}$
8: **return** Δ

Algorithm 4 gets a bounding box covering the entire road network. Using the grid setup (ψ_x, ψ_y) it computes the length and width of urban cells, and stores the partitioned road network into Δ. The computational complexity of Partition is $\Theta(1)$.

Algorithm 5 presents the strategy to translate the real mobility coordinates to grid coordinates.

Δ: Parameters of the grid;

T: Mobility trace.

Algorithm 5 Translate Real Coordinates To Grid Coordinates.

Input: Δ, T;
Output: G (mobility trace converted to grid coordinates);
1: $G \leftarrow \emptyset$
2: **for all** $\tau \in T$ **do** ▷ convert real traces to grid(x,y)
3: $x \leftarrow floor((\tau.x - \Delta.min\_\phi_x)/\Delta.size\_\phi_x)$
4: $y \leftarrow floor((\tau.y - \Delta.min\_\phi_y)/\Delta.size\_\phi_y)$
5: $G \leftarrow G \cup \{\tau.time, x, y\}$
6: **end for**
7: **return** G

Algorithm 5 converts the mobility trace (T) from real coordinates to grid urban cells. The computational complexity of Algorithm 5 is $\Theta(T)$ since it requires processing the entire mobility trace, and for each trace entry we have a fixed cost.

Algorithm 6 presents the strategy to compute the density of vehicles per urban cells. Below we describe input parameters:

Δ: Parameters of grid;

T: Mobility trace.

Algorithm 6 Compute Density of Vehicles in Grid.

Input: Δ, T;

Output: M (number of vehicles per urban cell);

```
1: for all i, j do
2:     M_{i,j} ← 0
3: end for
4: for all τ ∈ T do
5:     increase(M_{τ.x,τ.y})
6: end for
7: return M
```

Algorithm 6 initializes a matrix containing the density of vehicles per cell, counts the number of vehicles per urban cell, and returns the density of vehicles (M). The computational complexity of Algorithm 6 is $\Theta(T + M)$ because we have to process the matrix of locations (M) and the entire mobility trace (T). For most of the situations, T >> M.

Algorithm 7 presents the strategy to compute stochastic matrix (required for PMCP). Any vehicle has just four possible movements: (a) move up; (b) move down; (c) move left; (d) move right. Thus, each $P_{i,j}$ holds the number of vehicles performing each one of the four possible movements: $P^u_{i,j}$; $P^d_{i,j}$; $P^l_{i,j}$; $P^r_{i,j}$.

Algorithm 7 receives as input:

Δ: Parameters of the grid;

T: Mobility trace;

M: Number of vehicles per urban cell.

Algorithm 7 Compute Migration Ratios in Grid.

Input: Δ, T, M;

Output: P (stochastic matrix);

1: **for all** i, j **do**
2: $P_{i,j} \leftarrow \{0, 0, 0, 0\}$
3: **end for**
4: **for all** $\tau \in T$ **do**
5: $h \leftarrow \tau.next()$
6: **if** $(\tau.x = h.x)$ and $(\tau.y > h.y)$ **then**
7: $increase(P^{l}_{\tau.x,\tau.y})$
8: **else if** $(\tau.x = h.x)$ and $(\tau.y < h.y)$ **then**
9: $increase(P^{r}_{\tau.x,\tau.y})$
10: **else if** $(\tau.x > h.x)$ and $(\tau.y = h.y)$ **then**
11: $increase(P^{d}_{\tau.x,\tau.y})$
12: **else if** $(\tau.x < h.x)$ and $(\tau[i].y = h.y)$ **then**
13: $increase(P^{u}_{\tau.x,\tau.y})$
14: **end if**
15: **end for**
16: **for all** i, j **do**
17: **if** $(M_{i,j} \neq 0)$ **then**
18: $P^{u}_{i,j} \leftarrow P^{u}_{i,j}$ / $M[i][j]$
19: $P^{d}_{i,j} \leftarrow P^{d}_{i,j}$ / $M[i][j]$
20: $P^{l}_{i,j} \leftarrow P^{l}_{i,j}$ / $M[i][j]$
21: $P^{r}_{i,j} \leftarrow P^{r}_{i,j}$ / $M[i][j]$
22: **end if**
23: **end for**
24: **return** P

Algorithm 7 initializes the migration ratios (stochastic matrix). Then, it counts the number of vehicles moving changing urban cells, and computes the likelihoods of possible moves (up, down, left, and right). Finally, it returns the migration ratios (stochastic matrix). The complexity of Algorithm 7 is $\Theta(P + T + M)$. Note that $|P| = |M|$. Thus, complexity can be expressed as $\Theta(T + M)$.

After partitioning the road network using Algorithms 4-7 we may apply the PMCP-based and MCP-based approaches to place the roadside units in the α-better urban cells. Table 4.1 indicates the respective computational complexity of each algorithm. Line (g) presents the complexity of MCP-based plus partition technique as

$\Theta(\alpha.M.\log\ M+T+M)$. When the size of the mobility trace ($|T|$) is $\gg$ than the size of the grid $|M|$, we can express the complexity as $\Theta(T)$. Line (h) presents the complexity of the PMCP-based approach plus the partition technique as $\Theta(\alpha.M^2 + T + 2.M)$. Following the same reasoning, when $|T| \gg |M|$ the complexity becomes $\Theta(T)$. Line (i) presents the complexity of the brute force implementation $C_\alpha^{|M|}$ when we evaluate the roadside units in every possible positions of the grid.

Table 4.1. Computational Complexity.

Line	Method	Complexity
(a)	Partition Road Net.	$\Theta(1)$
(b)	Translate Coordinates	$\Theta(T)$
(c)	Density of Vehicles	$\Theta(T+M)$
(d)	Turning Matrix	$\Theta(T+M)$
(e)	MCP-based	$\Theta(\alpha.M.\log\ M)$
(f)	PMCP-based	$\Theta(\alpha.M^2.\log\ M)$
(g)	MCP-based+Partition (1)	(a)+(b)+(c)+(e)
	MCP-based+Partition (2)	$\Theta(\alpha.M.\log\ M+T+M)$
	MCP-based+Partition (3)	$\Theta(T)$, if $\|T\| \gg \|M\|$
(h)	PMCP-based+Partition (1)	(a)+(b)+(c)+(d)+(f)
	PMCP-based+Partition (2)	$\Theta(\alpha.M^2+T+M)$
	PMCP-based+Partition (3)	$\Theta(T)$, if $\|T\| \gg \|M\|$
(i)	Brute force	$C_\alpha^{\|M\|} = \frac{M!}{\alpha!(M-\alpha)!}$

Where:
M is the number of partitions;
T is the number of entries in the vehicular trace file;
α is the number of available roadside units.

4.3 Adapting the Flow Projection to Urban Cells

The projection of the flow has the purpose of discovering the number of vehicles that will reach a given urban cell. Basically, we count the number of vehicles at each one of the preceding urban cells, and we obtain the projected number of vehicles using stochastic matrix P (migration ratios). The process is quite simple: suppose that we have just inserted a new roadside unit within the road network. After inserting the new roadside unit, we employ the projection of the flow to infer the expected number

vehicles arriving from all urban cells that will cross this newly deployed roadside unit. The identified vehicles are removed from the model since they will reach this new added roadside unit soon. In order to compute the projection $P_{i,j}$ for urban cell $M_{i,j}$, we propose the use of four different projections (Figure 4.3):

- Vehicles moving left-to-right towards target urban cell;
- Vehicles moving right-to-left towards target urban cell;
- Vehicles moving bottom-up towards target urban cell;
- Vehicles moving top-down towards target urban cell.

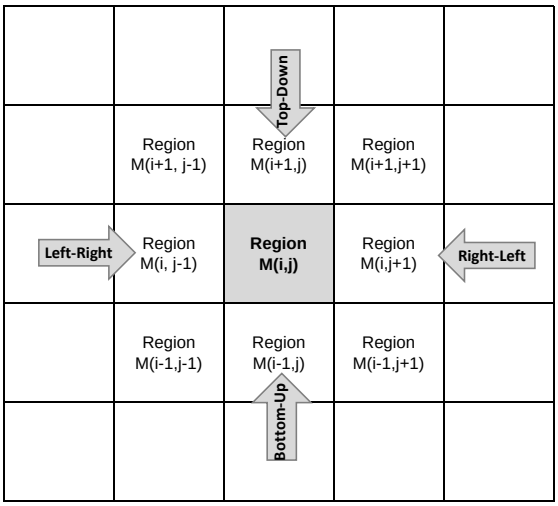


Figure 4.3. Urban cells composing a road network: projection of the flow identifies vehicles moving in all four directions.

Left arrow in Figure 4.3 represents vehicles moving left-to-right towards urban cell $M_{i,j}$. Expected number of vehicles ($LR_{i,j}$) arriving from urban cells $M_{i,j-1}$, $M_{i,j-2}$, ... , $M_{i,1}$ to $M_{i,j}$ is:

$$LR_{i,j} = \sum_{k=0}^{j-1}(UncoveredVehicles(M_{i,k}).\prod_{l=k}^{j-1} P_{i,l}) \tag{4.2}$$

Equation 4.2 is very similar to Equation 3.3. Similarly, we derive Equations4.3-4.6 considering the grid scenario. Such Equations are presented below.

$$RL_{i,j} = \sum^{\forall k>j}(UncoveredVehicles(M_{i,k}).\prod_{l=j+1}^{k} P_{i,l}) \tag{4.3}$$

$$BU_{i,j} = \sum_{k=0}^{i-1}(UncoveredVehicles(M_{k,j}).\prod_{l=k}^{i-1} P_{j,k}) \tag{4.4}$$

$$TD_{i,j} = \sum^{\forall k>i}(UncoveredVehicles(M_{k,j}).\prod_{l=i+1}^{k} P_{j,k}) \tag{4.5}$$

Therefore, total projected flow ($PF_{i,j}$) is computed by adding $LR_{i,j}$, $RL_{i,j}$, $BU_{i,j}$ and $TD_{i,j}$ to the number of uncovered vehicles inside urban cell $M_{i,j}$, as Equation 3.7 shows.

$$\begin{aligned} PF_{i,j} = &UncoveredVehicles(M_{i,j})+ \\ &LR_{i,j} + RL_{i,j} + BU_{i,j} + TD_{i,j} \end{aligned} \tag{4.6}$$

4.4 Methodology

According to our proposal, the deployment of roadside units is a two-stage process:

(a) Partition the road network;

(b) Apply PMCP-based approach to select the α-better cells.

We consider the map presented in Figure 4.2(a) as our road network. We have elected the simulator 'the ONE' (Keränen et al., 2009) for generating our mobility traces. Although 'the ONE' is not a vehicular mobility tool, it does the job of moving the nodes along the map according to simulation parameters. As we are concerned just about the number of vehicles reaching roadside units in low/medium traffic conditions, we have considered the simulator enough for the purposes of this simulation. We have simulated 50 samples of 10 000s (each) of traffic using 60 vehicles with speeds ranging from 8m/s to 17m/s, moving between source and destination using always the shortest route. When a vehicle reaches the destination, it sleeps for a random time over 100s to 200s (uniform distribution), then it picks another random destination and restarts trip. Our scenarios do not contain attraction areas in order to generate a fuzzy traffic flow. The fuzzy traffic is intentional as it allows us to perform a worst-case analysis of the both strategies performance.

We propose two sets of experiments to evaluate the effects of applying the **partition** technique to support the deployment of roadside units: S_1 and S_2.

In Experiments S_1 (Section 4.5) we always select a constant number of roadside units for several grid setups (from 20x20 up to 100x100 urban cells). This evaluation

is justified by our need to perform a preliminarily comparison of the PMCP-based and MCP-based approaches considering a partitioned road network. Since we are always selecting a constant number of α cells to receive the roadside units, the grid containing the largest urban cells will always incur in better results, after all, its α cells correspond to a largest portion of the road network receiving roadside units (as we detail in the next paragraphs, we assume that each selected cell is fully covered; otherwise our results may be affected by the quality of intra-cell deployment strategies). In Experiments S_1 we focus on comparing how the selection of urban cells is carried out by the PMCP-based and MCP-based approaches (and not comparing different grid setups).

In Experiments S_2 (Section 4.6) we compare distinct grid setups. We reevaluate the same scenario, but now selecting a variable number of cells. Besides that, we also evaluate the performance of the deployment using grid setups up to 800x800 urban cells. Experiments S_2 complement S_1: while in S_1 we are focused on evaluating the performance of MCP-based x PMCP-based using partitions, in S_2 we clarify not so obvious aspects of the partition process. Below we characterize the flow of vehicles used during our experiments.

4.4.1 Vehicles Flow Characterization

Figure 4.4 presents the distance traveled by each vehicle.

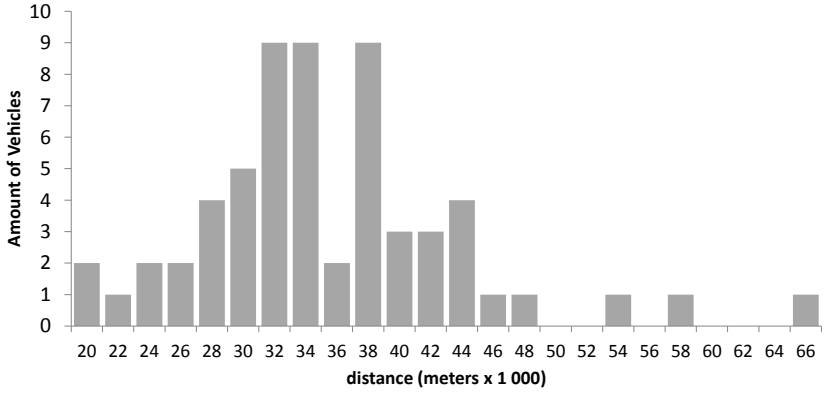


Figure 4.4. Distance traveled by each vehicle during 10 000s of simulation.

Figure 4.5 shows the routes of vehicles. Shadows indicate the amount of vehicles, with major roads presenting higher traffic.

Figure 4.5. Map demonstrates the level of traffic in every road. Shadows indicate popularity of the road.

4.5 Experiments S_1: PMCP-based and MCP-based over Partitions

In this Section we present experiments comparing the performance of the MCP-based and PMCP-based approaches over partitioned and non-partitioned areas. Our goal is to reveal weather the PMCP-based approach is able to select better urban cells than the MCP-based one. We assume each urban cell to be indivisible. Thus, every cell containing a roadside unit is always fully covered in order that our results do not depend on intra-cell deployment strategies. For every experiment we select the five-better cells of each grid setup. The number five was chosen arbitrarily. In fact, any $n > 0$ and $n \ll 100$ would serve our purposes, where 100 is the quantity of cells within the smallest grid setup (10x10).

4.5.1 Deployment Efficiency

Figure 4.6 plots the percentage of covered urban area and the number of trips experiencing at least one V2I contact opportunity. The x-axis indicates the covered area (i.e., the percentage of urban cells fully covered by roadside units), while the y-axis indicates the number of trips. The results demonstrate that by covering 5% of the urban area we achieve 90% of the vehicles experiencing at least one V2I contact opportunity when we deploy roadside units according to a PMCP-based approach. When we employ a MCP-based approach, we achieve 82.8% of the trips.

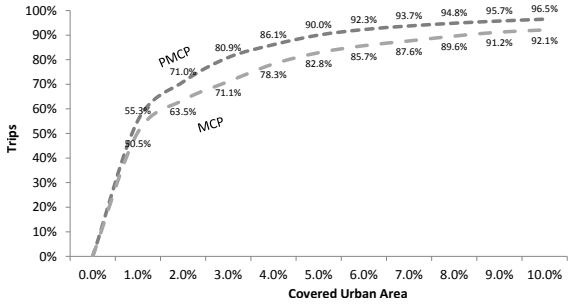


Figure 4.6. Area x Trips: The x-axis indicates the covered area (i.e., the percentage of urban cells fully covered by roadside units), while the y-axis indicates the number of trips.

4.5.2 Fairness in Contacts Opportunities

In this experiment we present the number of distinct vehicles crossing the five selected cells. Our goal is to reveal how each deployment algorithm performs in terms of fairness of V2I contact opportunities. In Figure 4.7, the x-axis indicates the grid setup. The entry 'No Grid' shows the results when we select the location of each roadside unit without partitions. We can notice that the percentage of distinct-and-covered vehicles decreases as the grid size increases. Such issue happens because the size of each cell also decreases (remember that we assume full-coverage of an entire cell). A typical cell in the 10x10-grid has area equal to 1 200 000m^2. On the other hand, a typical cell in the 80x80-grid is 18 750m^2. So, cells of the 10x10-grid are 64 times bigger than cells of the 80x80-grid causing the descending aspect of the plot in Figure 4.7.

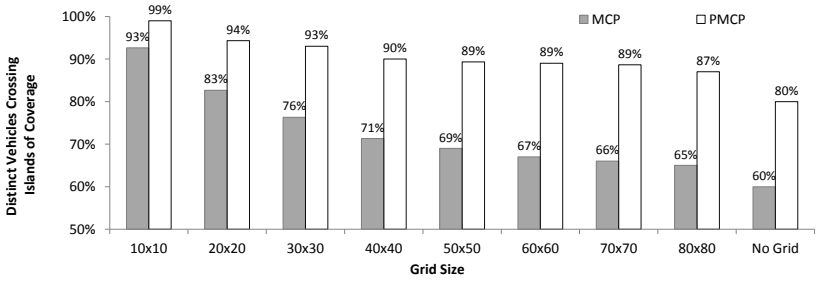


Figure 4.7. Fairness in contacts opportunities: The plot shows the amount of distinct vehicles crossing any of the five most important partitions during the simulation. The x-axis indicates the grid setup. The y-axis indicates the number of distinct vehicles as a percentage of total vehicles. The partitions are selected using MCP-based and PMCP-based. 'No Grid' identifies the result for a non-partitioned road network.

PMCP-based outperforms MCP-based. Since PMCP-based considers the probability of each vehicle to reach each partition, PMCP-based achieves a more

complete evaluation of the vehicles mobility, allowing the PMCP-based approach to choose better partitions. Non-partitioned road network ('No Grid') receives less vehicles because the covered area is restricted to the range of the roadside units. This dataset matters because it indicates that evaluating the whole road network without **partitions** does not bring an outstanding quality of deployment. However, the computational cost of evaluating a non-partitioned area increases exponentially. While the current experiment shows the number of distinct vehicles reaching the selected cells considering the full simulation, the next one reveals how many times vehicles reach the selected urban cells.

4.5.3 Cumulative Distinct Contacts Over Time

In this experiment we are interested in the number of times that vehicles enter the selected urban cells (roadside units cells). In Figure 4.8, the x-axis represents segmented time intervals. The y-axis indicates the cumulative number of contacts. PMCP-based is represented by solid dashes, and MCP-based by dotted dashes. When considering the entire simulation (x-axis=1-10), PMCP-based offers 407-443 contacts, depending on the grid setup (from 20x20 to 100x100), while MCP-based offers 205-280 contacts.

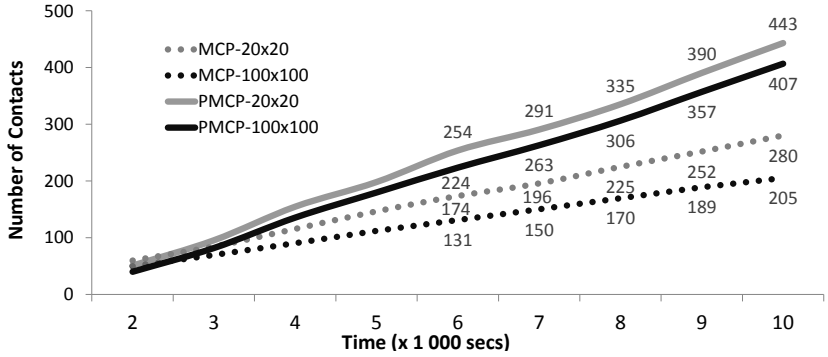


Figure 4.8. Number of V2I Contacts: The x-axis indicates incremental time intervals. The y-axis indicates distinct contacts.

Figure 4.9 indicates the cumulative average number of contacts and the standard deviation. The average is computed over the number of contacts for grid setups 10x10, 20x20, 30x30,..., 100x100. The figure highlights the main difference between the PMCP-based and the MCP-based approaches: during initial moments of the simulation (until 2 000s), MCP-based shows a better performance. Such happens because MCP-based selects the densest urban cells.

When the simulation reaches 3 000s, the PMCP-based overcomes MCP-based.

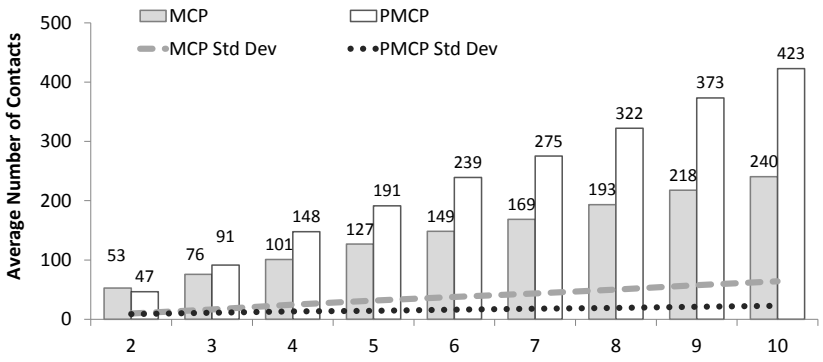


Figure 4.9. Average contacts for distinct grid setups. The x-axis indicates time interval. The y-axis indicates contacts.

4.5.4 Vehicle-to-Infrastructure Contact Time

Here we present the V2I contact time. In Figure 4.10, x-axis indicates the grid setup, while the y-axis indicates the total residence time of vehicles within roadside units.

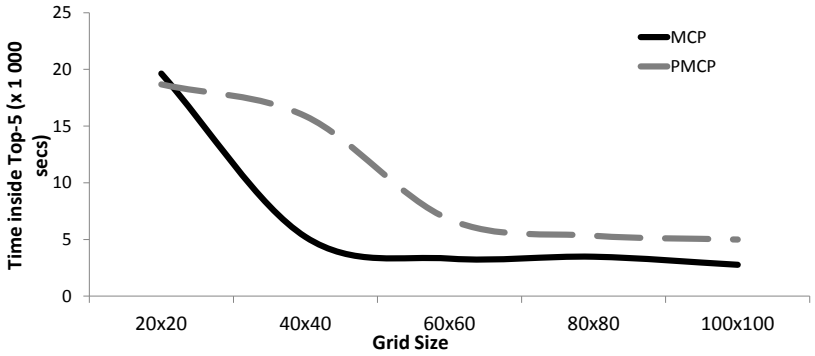


Figure 4.10. Vehicle-to-infrastructure contact time: x-axis indicates the grid setup, while the y-axis indicates the total residence time.

We notice that the 20x20-grid is less affected by MCP-based and PMCP-based for a simple reason: very large cells of the 20x20-grid attenuate the probability of vehicles' migrations between adjacent urban cells. Thus, the projection of flow is not strong enough to change the selection of urban cells proposed by MCP. On the other hand, as we increase the number of grid cells, we reduce the area occupied by the roadside units cells. Smaller areas tend to hold fewer vehicles, which causes the reduction in the residence time shown in the Figure. The Figure also shows that each deployment algorithm reacts differently to variations in the size of the urban cells. Finally, the Figure also demonstrates that the residence time becomes stable for grid setups over 60x60, highlighting the possible existence of an upper bound in the number of partitions for a given road network.

Standard deviation of PMCP-based is presented in Figure 4.11.

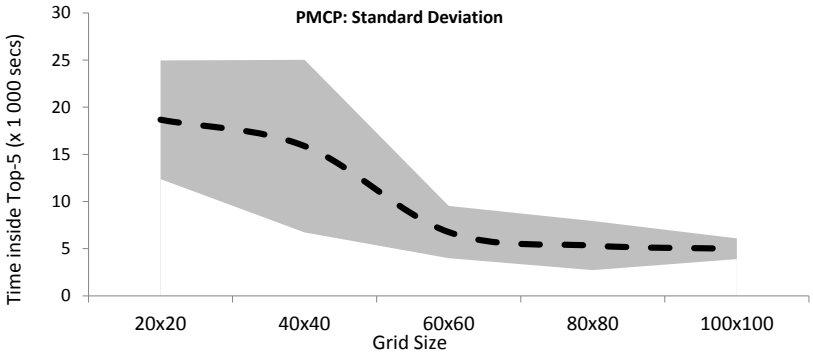


Figure 4.11. PMCP: Standard deviation for V2I contact time.

Figure 4.12 presents standard deviation for MCP.

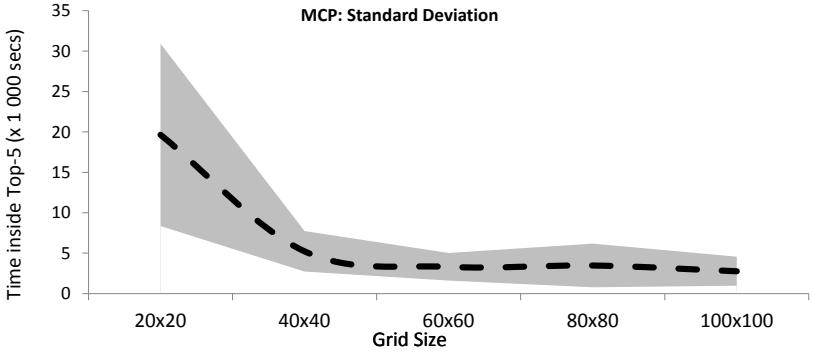


Figure 4.12. MCP: Standard deviation for V2I contact time.

4.5.5 Cumulative Contact Time

Now we evaluate the time duration of vehicle-to-infrastructure contacts. In Figure 4.13, x-axis indicates cumulative time intervals, while y-axis indicates contact time of all active vehicles. Values shown in plot represent average contact time when considering ten grid setups (10x10, 20x20, ..., 100x100). Contact time is almost linear overtime. Selection of cells made by PMCP-based offers more vehicle-to-infrastructure contact time.

4.5.6 Dynamic Deployment of Roadside Units

When we partition a road network, we reduce the demand for computing resources: instead of analyzing every possible point in a 2-D plane, we concentrate our analysis

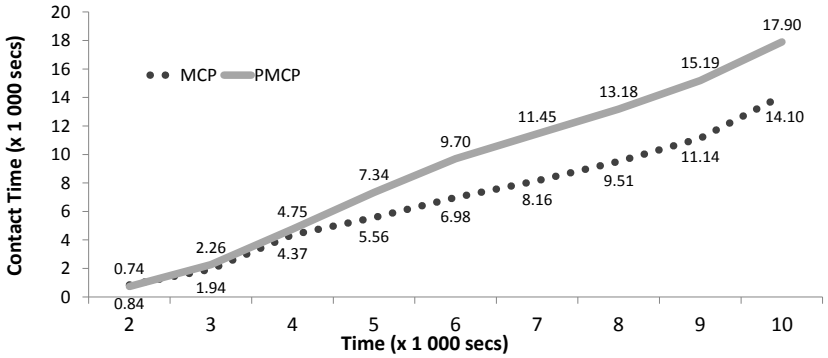


Figure 4.13. Average residence time (contact time): the x-axis indicates the time interval, while the y-axis indicates the V2I contact time.

on urban cells covering the road network, which gives us the opportunity to provide real-time services by decreasing the granularity of the road network in order to reduce the computation time. At a first glance, the discussion of the dynamic allocation of roadside units seems awkward. However, in order to address traffic fluctuation, we may rely on a virtual-and-mobile set of roadside units allocation of roadside units. Besides that, the allocation of roadside units is a metaphor to a wider class of resources allocation problem. For instance, real-time mechanisms may: (i) arrange resources for maximum efficiency in the network; (ii) manage the content distribution (music, video, games); (iii) support caching mechanisms (transferring contents in the right measure to vehicles during the trip), among several others.

Figure 4.14 compares the static deployment and dynamic deployment of roadside units for PMCP-based and MCP-based:Static deployment considers first 500s of traffic to select the roadside units cells. Dynamic deployment also considers the first 500s of traffic to choose the roadside units cells, but it recalculates the roadside units cells every 1 000s. We compare the efficiency of both methods by evaluating the number of vehicle-to-infrastructure contacts.

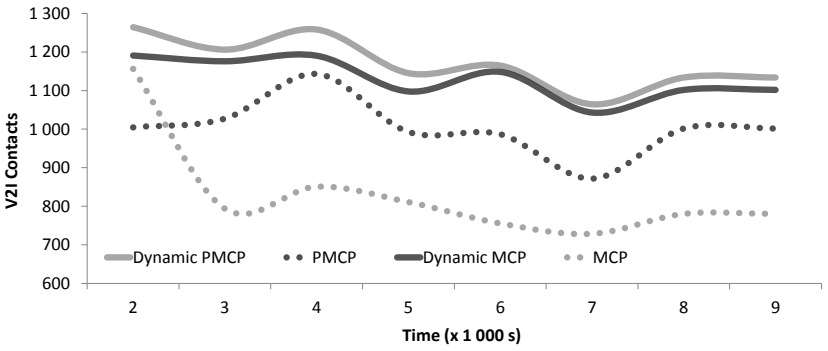


Figure 4.14. V2I Contact Time Considering Dynamic Allocation.

The interval time of 1 000s is chosen arbitrarily: our goal is just to demonstrate that even a naive dynamic strategy is able to improve resources allocation when compared to static methods. The experiment demonstrates that dynamic deployment overcomes static ones. Dynamic MCP-based is similar to static PMCP-based: since the location of each roadside unit is constantly recomputed, Dynamic MCP-based is able to reconfigure the roadside units according to the vehicles flow. On the other hand, the PMCP-based presents less benefits since it already employs the mobility model to allocate the roadside units. Figure 4.15 shows vehicle-to-infrastructure contact time when we vary the grid setup. In this experiment, Dynamic MCP-based overcomes static PMCP-based.

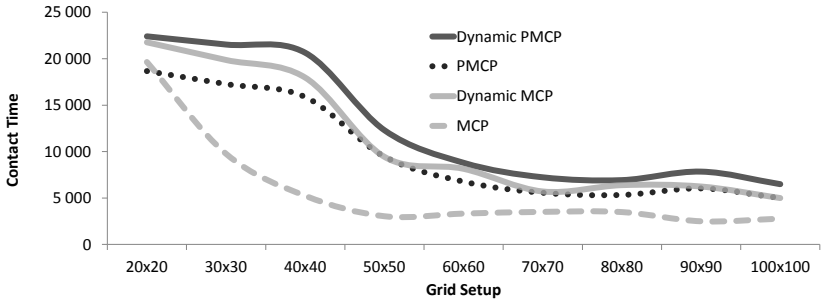


Figure 4.15. V2I Contact Time: The x-axis shows the time interval. The y-axis shows the V2I contact time.

4.5.7 Dynamic MCP-based x Static MCP-based

Now we analyze each interval of 1 000s of the MCP-based strategy. We restrict our analysis to MCP-based because it shows more discrepancy between the static and the dynamic results. Our goal is to reveal how the static/dynamic schemes perform on time windows. Figure 4.16 shows the contact time for both schemes: the x-axis indicates the time window, while y-axis shows the contact time. The dynamic algorithm shows approximately 800s of residence time, while the static one achieves approximately 500s. The dotted lines refer to the secondary y-axis and indicate the deviation for both samples.

4.5.8 Remarks

These experiments have characterized the selection of urban cells when using the PMCP-based and the MCP-based approaches. The main conclusions are: (i) PMCP-based improves the fairness in contacts over MCP-based; (ii) Larger cells are

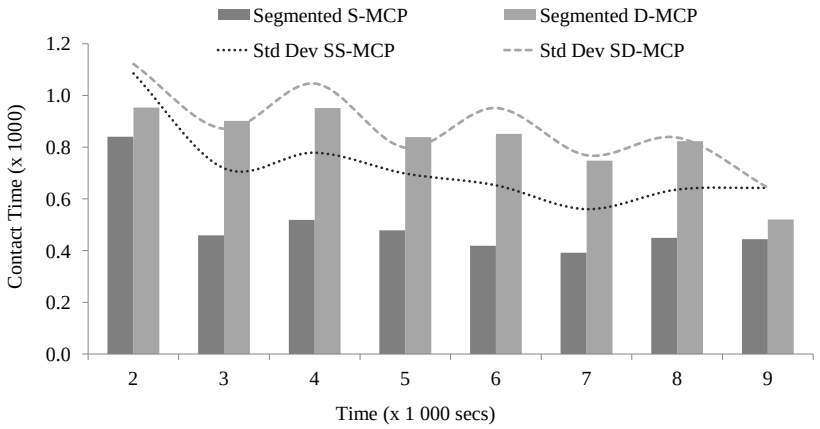


Figure 4.16. Evaluation segmented in time: x-axis indicates the time window. The y-axis indicates the V2I contact time.

less affected by PMCP-based because they present less migration events; (iii) Because traffic fluctuates we may rely on dynamic methods to improve the resources allocation. Whenever we recalculate the position of each roadside unit every 1 000s, we improve the vehicle-to-infrastructure contact time in almost 60%; (iv) Dynamic MCP-based achieves a closer performance to a PMCP-based. (v) PMCP-based presents less benefit because it uses the projection of the flow to allocate the roadside units. In the next Section we present the set of experiments S_2.

4.6 Experiments S_2: Comparing Grid Setups

The experiments S_2 present a complementary evaluation of the traffic scenario analyzed in S_1. The main difference is that now we 'cover' a fixed percentage of the entire road network for grid setups ranging from 20x20 up to 800x800. Initially we have to define the percentage of area to be covered: very large covered areas attenuate differences when we vary grid setup (i.e., all vehicles are covered during the trip for all grid setups), possibly hiding relevant aspects of the partition technique. A similar behavior may happen if we 'cover' a very small fraction of the road network.

Supported by our results presented in Section 3.5.1 and Section 4.5.2, we cover 5% of the road network. Recall that Section 3.5.1 reports that, by covering 6.2% of the road network, the PMCP-based approach achieves approximately 100% of vehicle-to-infrastructure contacts, while MCP achieves 88%; and, Section 4.5.2 reports that, by selecting five cells from a 10x10-grid, PMCP-based achieves 99% of contacts, while MCP-based achieves 93%.

Table 4.2 presents the grid setups, total number of cells of each setup, and number of cells receiving roadside units in order to cover 5% of the entire road network.

Table 4.2. Grid Setups.

Setup	20x20	40x40	60x60	80x80	100x100	200x200	300x300	400x400	800x800
Urban Cells	400	1 600	3 600	6 400	10 000	40 000	90 000	160 000	640 000
Roadside Units	20	80	180	320	500	2 000	4 500	8 000	32 000

4.6.1 Over-Partitioning

In this experiment we demonstrate the total V2I contact time when we vary the grid setup from 20x20 (400 urban cells) up to 800x800 (640 000 urban cells). In Figure 4.17, the x-axis indicates grid setup for MCP-based (left-half) and PMCP-based (right-half), while the y-axis indicates the V2I contact time (×1 000s). The Figure clearly demonstrates that over-partitioning does not improve the deployment efficiency indefinitely. Partitioning such road network into 800x800 urban cells implies in defining 640 000 urban cells, where each cell is approximately 13m x 13m.

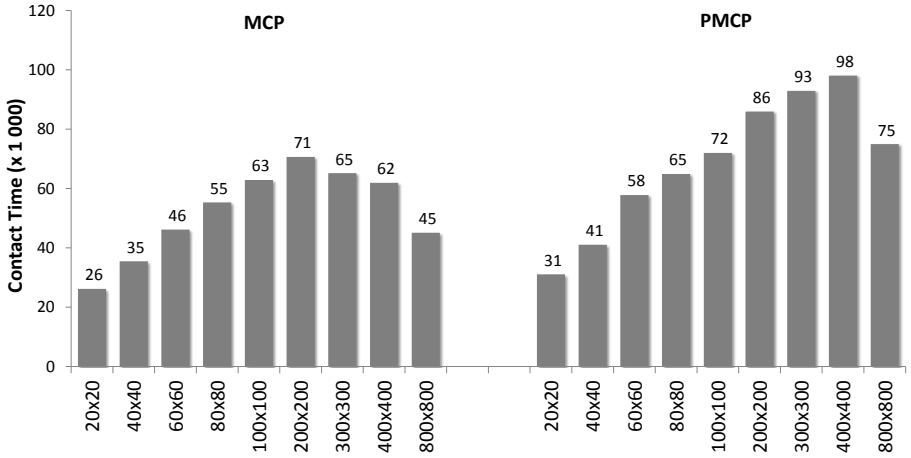


Figure 4.17. Total V2I Contact Time for Distinct Grid Setups: The x-axis indicates the grid setup (from 20x20 to 800x800) for both MCP and PMCP. The y-axis indicates the sum of V2I contact time for all vehicles (x 1 000s).

Very small cells incur in distortions and unrealistic behaviors (such as a roadside unit covering just one lane in a two-lane street) that degenerates the performance of the deployment. Such issue is a side effect of our assumption that a roadside unit covers an entire urban cell. Figure 4.18 illustrates this.

Given a grid setup, PMCP-based approach always offers more V2I contact time. The maximum contact time for MCP-based is achieved when the grid setup is 200x200

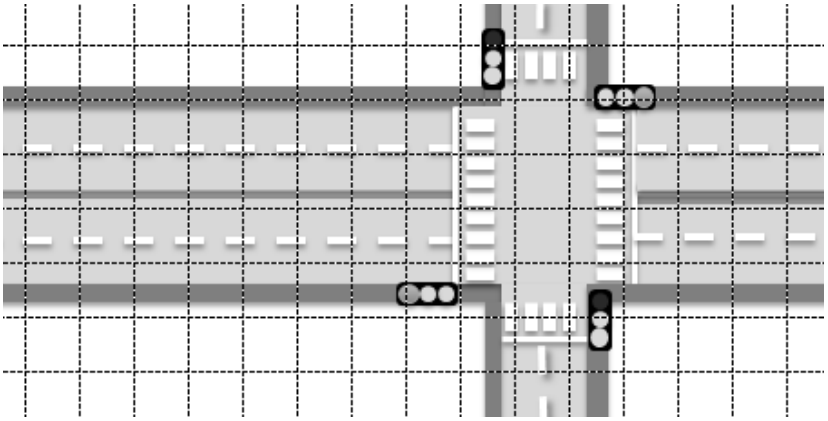


Figure 4.18. Very small cells covering small pieces of a lane.

(71 000s). After such threshold, the contact time decreases when we increase the number of grid cells. On the other hand, the PMCP-based requires a 100x100-grid in order to achieve more contact time than the MCP-based in a 200x200-grid. And, unlike the MCP-based, the performance of the PMCP-based continues to grow until grid setup is 400x400. The aspect of mobility incorporated by PMCP has a significant impact over the deployment performance.

4.6.2 Characterizing the Contact Time

In this experiment we analyze the total V2I contact time segmented into time windows of 1 000s. Our goal is to analyze how both deployment strategies (MCP-based and PMCP-based) perform over time. Notice that we are dealing with static deployment: both deployment strategies analyze 500s of initial traffic in order to allocate roadside units. Since PMCP-based assumes migration ratios as the most reliable source of information, the PMCP-based computes the migration ratios employing the whole sample (10 000s).

Figure 4.19 indicates the contact time for both MCP-based and PMCP-based considering a road network partitioned into a 20x20-grid. The x-axis indicates the time interval, while the y-axis indicates the total contact time. In the first interval (1 000s), both strategies demonstrate high performance, after all, their estimates of traffic are very accurate. When x=2 000s the Figure indicates a depression. Such depression is a characteristic of the trace file. During this time window, vehicles exhibit a different pattern of mobility, which lowers the performance of both deployments. After that, PMCP-based returns to a high level of performance, while MCP-based continues low profile. When x=6 000s we have another depression. Just like the previous one, such depression is more intensive in MCP-based.

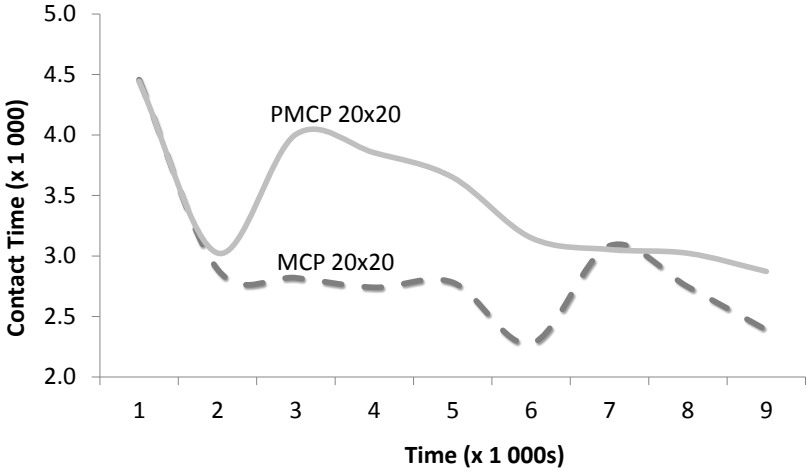


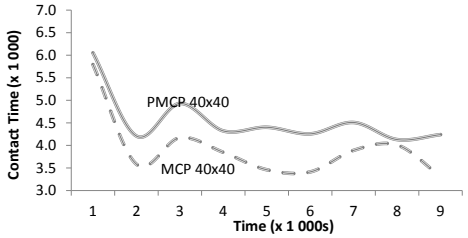
Figure 4.19. V2I Contact Time: 20x20-grid. The x-axis indicates the time window, while the y-axis indicates the total contact time.

In Figure 4.20(a) we analyze the results of MCP-based and PMCP-based considering a 40x40-grid. As we can notice, both strategies demonstrate a similar shape, and the PMCP-based presents a higher V2I contact time than MCP-based. Again, the depression of x=6 000s causes a stronger impact in the MCP-based approach. The experiment reveals that the grid setup influences the performance of the deployment. In order to complement this evaluation, the Figure 4.20 presents the total contact time for several grid setups. Table 4.3 presents the statistical description of the total V2I contact time for different grid setups. Typically, PMCP presents higher V2I contact time with lower deviation indicating that PMCP improves the deployment performance. Figure 4.21 presents the contact time for all 60 vehicles of PMCP-based.

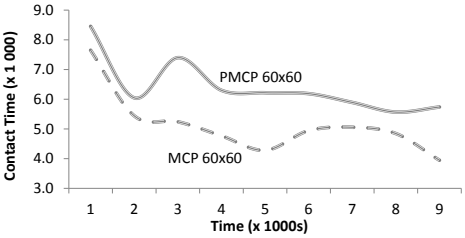
4.6.3 Crossed Roadside Units

Here we analyze the number of crossed roadside units per vehicle. As mentioned before, the simulation has a total of 60 vehicles traveling to random destinations always selecting the shortest route. In Figure 4.22, the x-axis denotes the vehicle's ID, while the y-axis indicates the number of crossed roadside units per vehicle. Notice that vehicles are sorted in descending order according to the number of crossed roadside units.

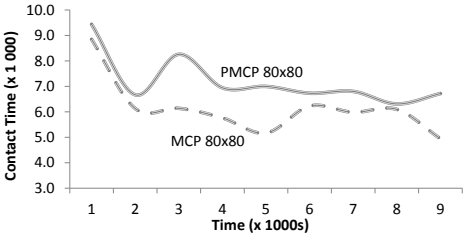
Figure 4.22(a) presents results considering a 20x20-grid where exactly 20 cells are selected to receive roadside units. Vehicle #1 crosses 32 roadside units during entire simulation (10 000s). Vehicle #60 crosses approximately five roadside units. Again, PMCP-based demonstrates a better performance. Figure 4.22(b) considers



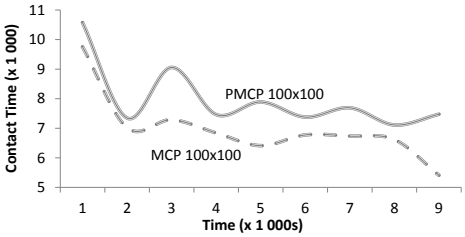
(a) 40x40.



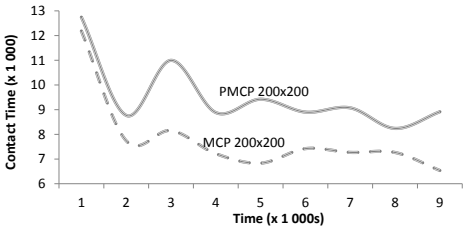
(b) 60x60.



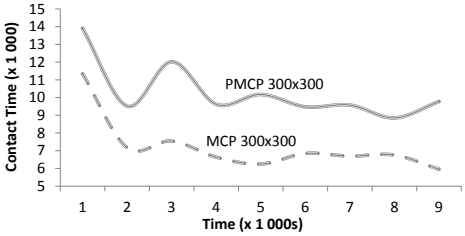
(c) 80x80.



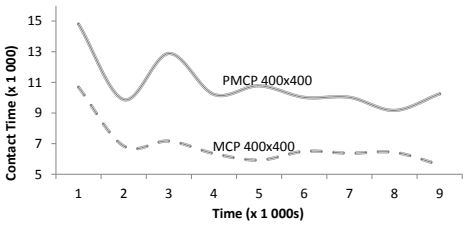
(d) 100x100.



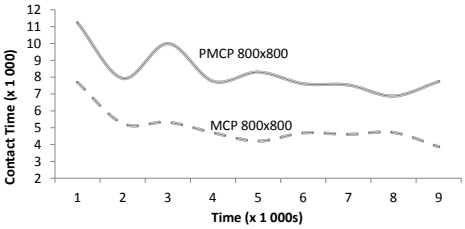
(e) 200x200.



(f) 300x300.



(g) 400x400.



(h) Contact Time: 800x800-grid.

Figure 4.20. Contact Time for Several Grid Setups.

a 100x100-grid having 500 roadside units. Figure 4.22(c) presents a 200x200-grid with 2 000 roadside units, while Figure 4.22(d) presents a 800x800-grid with 32 000 roadside units. Figure 4.23 plots the standard deviation of roadside units per vehicle. Table 4.4 presents the statistical description of number of roadside units per vehicle when considering grid setups from 20x20 up to 800x800.

Table 4.3. Contact Time Characterization: Intervals of 1 000s.

	20x20	40x40	60x60	80x80	100x100	200x200	300x300	400x400	800x800
Average (x 1 000)									
PMCP	3.45	4.56	6.42	7.20	7.99	9.54	10.32	10.89	8.32
MCP	2.90	3.94	5.12	6.14	6.98	7.84	7.24	6.87	5.00
Median (x 1 000)									
PMCP	3.15	4.32	6.18	6.80	7.47	8.91	9.61	10.21	7.75
MCP	2.78	3.84	4.94	6.09	6.77	7.27	6.75	6.43	4.70
Minimum Value (x 1 000)									
PMCP	2.87	4.12	5.56	6.31	7.11	8.24	8.85	9.17	6.88
MCP	2.27	3.31	3.94	4.94	5.41	6.53	5.95	5.62	3.87
Maximum Value (x 1 000)									
PMCP	4.44	4.60	8.45	9.43	10.57	12.74	13.89	14.80	11.22
MCP	4.45	5.79	7.64	8.84	9.75	12.18	11.32	10.68	7.68
Standard Deviation									
PMCP	552	607	922	993	1 118	1 420	1 603	1 791	1 379
MCP	630	752	1 050	1 111	1 163	1 690	1 600	1 497	1 100
Confidence Interval (95%)									
PMCP	424	466	709	764	859	1 092	1 232	1 376	1 060
MCP	484	578	807	854	894	1 299	1 230	1 151	845

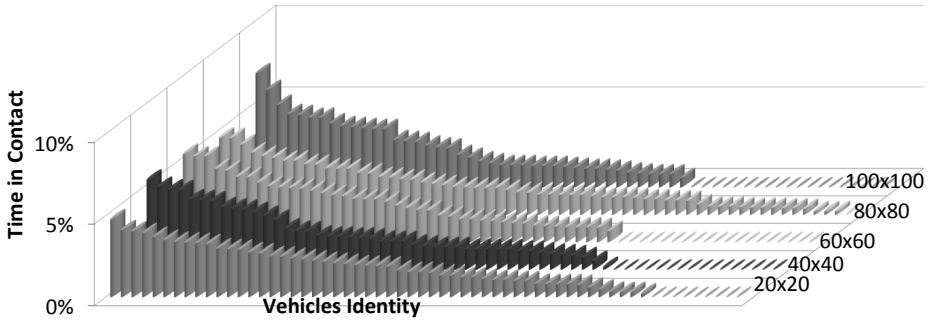


Figure 4.21. Vehicles Contact Time.

4.6.4 Comparison of Crossed Roadside Units

Our goal here is to compare the number of crossed roadside units for distinct grid setups. Such comparison is not trivial, after all each grid setup has a distinct number of roadside units. Thus, we propose the normalization of each sample to a Standard Gaussian Distribution $\mu = (0, 1)$. The normalized number of roadside units is computed as indicated in Equation 4.7:

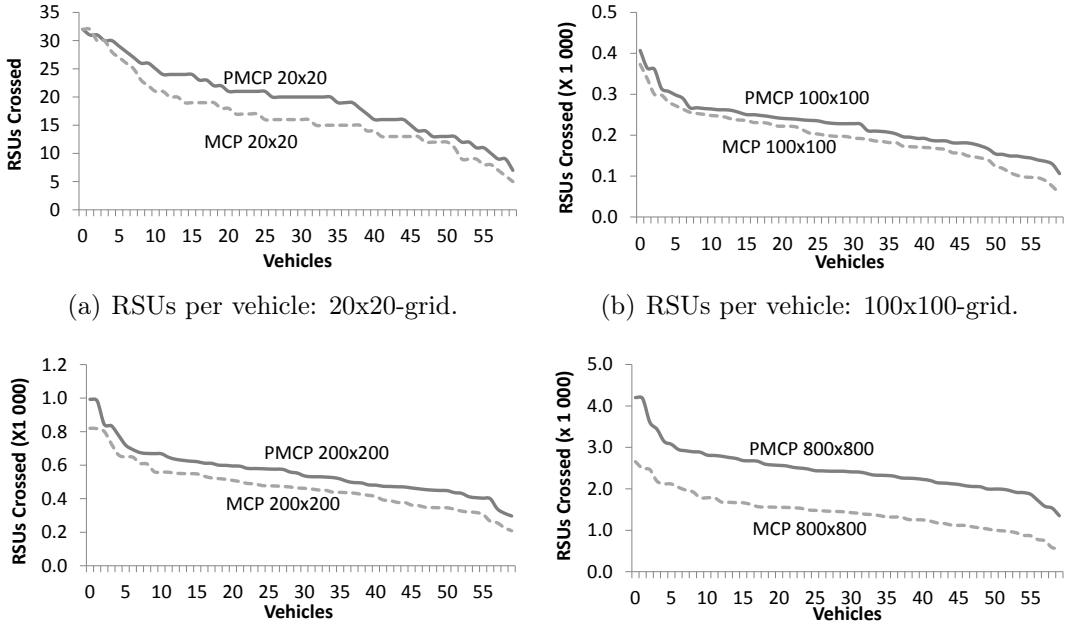
$$N_{\nu,g} = \frac{R_{\nu,g} - \overline{R_g}}{\sigma_g}, \forall \nu \in V \tag{4.7}$$

Where:

$N_{\nu,g}$= normalized number of roadside units crossed by vehicle ν in grid setup g;

$R_{\nu,g}$= number of roadside units crossed by vehicle ν in grid setup g;

$\overline{R_g}$= average number of roadside units crossed in grid setup g;



(a) RSUs per vehicle: 20x20-grid.

(b) RSUs per vehicle: 100x100-grid.

(c) RSUs per vehicle: 200x200-grid.

(d) RSUs per vehicle: 800x800-grid.

Figure 4.22. Number of RSUs per vehicle for several grid setups.

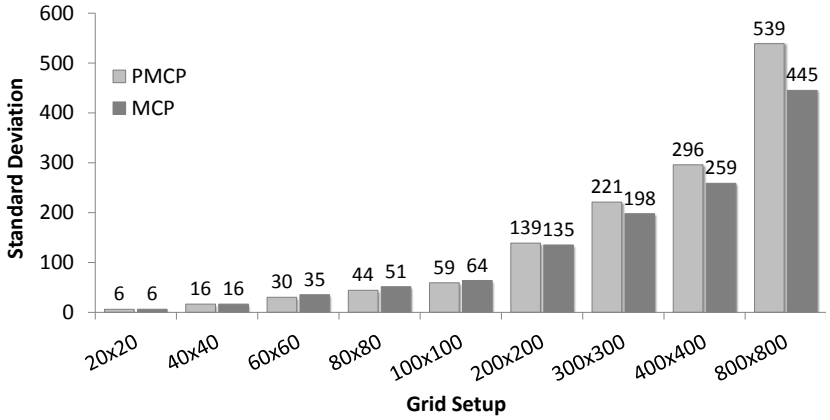


Figure 4.23. Standard Deviation: Roadside Units per Vehicle: The x-axis indicates the grid setup, while the y-axis indicates the standard deviation.

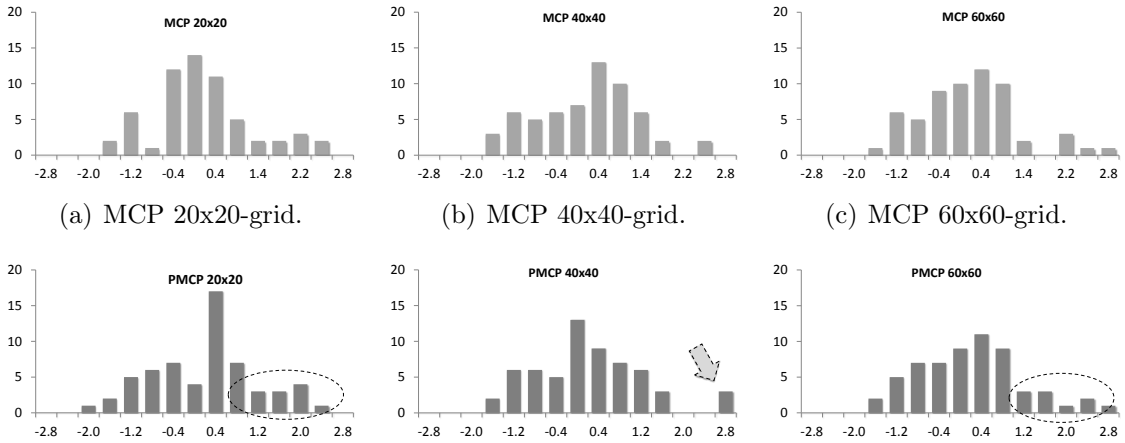
σ_g= standard deviation of number of roadside units crossed in grid setup g;
V= set of vehicles.

Figure 4.24 illustrates the Standard Gaussian of roadside units crossed for MCP-based (a-c) and PMCP-based (d-f) for grid setups ranging from 20x20 up to 60x60. Different grid setups incur in slightly distinct number of roadside units crossed

Table 4.4. Number of crossed roadside units per vehicle.

	20x20	40x40	60x60	80x80	100x100	200x200	300x300	400x400	800x800
Average RSUs crossed									
PMCP	19.6	55.3	104.9	160.2	221.9	559.2	933.9	1 349.3	2 454.8
MCP	16.6	46.6	87.2	136.4	194.7	469.7	662.3	851.2	1 439.4
Median									
PMCP	20.0	54.5	104.5	163.0	228.0	544.5	898.0	1 295.0	2 415.5
MCP	16.0	48.5	86.5	134.5	195.0	463.0	652.5	834.0	1 435.5
Minimum number of RSUs crossed									
PMCP	7	24	53	81	106	297	532	739	1 351
MCP	5	16	26	33	57	209	274	340	573
Maximum number of RSUs crossed									
PMCP	32	97	187	299	407	993	1 663	2 319	4 201
MCP	32	88	188	288	373	820	1 190	1 544	2 654
Standard Deviation									
PMCP	6.1	16.4	30.1	44.2	59.2	138.7	221.0	295.9	538.9
MCP	6.2	16.4	35.2	51.4	63.9	134.9	197.8	258.8	445.2
Confidence Interval (95%)									
PMCP	1.5	4.2	7.7	11.4	15.3	35.8	57.1	76.4	139.2
MCP	1.6	4.2	9.1	13.3	16.5	34.8	51.1	66.8	115.1

per vehicle. Arrows and circles highlight the differences between PMCP-based and MCP-based approaches: PMCP-based has more vehicles presenting a high number of contacts. Figure 4.25 presents the Standard Gaussian for grid setups 80x80, 100x100 and 200x200: just like the previous Figure, the PMCP-based increases the number of vehicles presenting a high number of V2I contacts. Figure 4.26 presents Standard Gaussian of crossed roadside units for grid setups 300x300, 400x400 and 800x800.

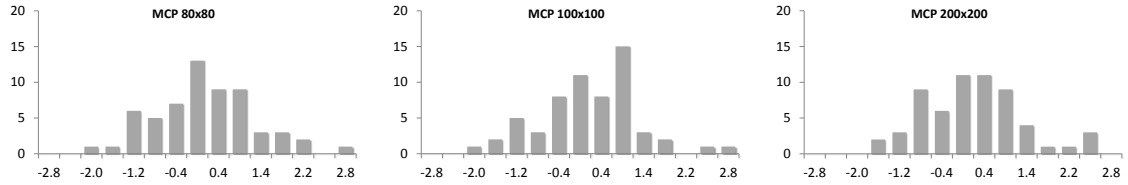


(a) MCP 20x20-grid. (b) MCP 40x40-grid. (c) MCP 60x60-grid.

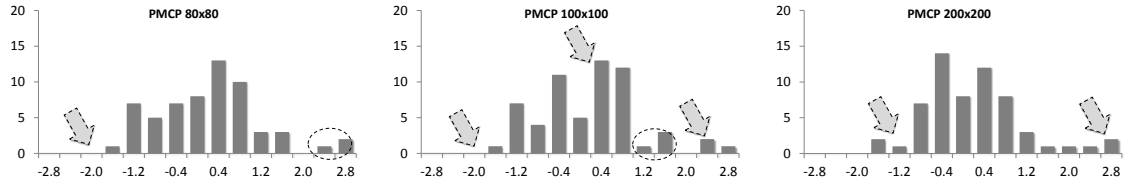
(d) PMCP 20x20-grid. (e) PMCP 40x40-grid. (f) PMCP 60x60-grid.

Figure 4.24. Crossed Roadside Units Normalized to Standard Gaussian: Grid Setups 20x20, 40x40, and 60x60.

Finally, we present another comparison between number of roadside units crossed for every vehicle composing our sample. In Figure 4.27, the x-axis indicates the vehicle's

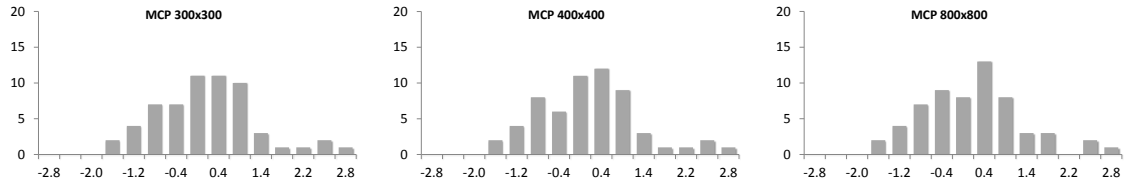


(a) MCP 80x80-grid. (b) MCP 100x100-grid. (c) MCP 200x200-grid.

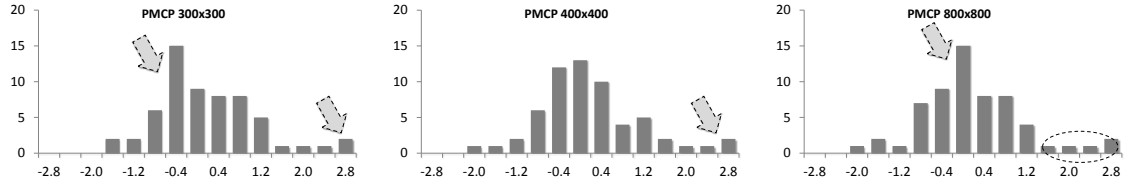


(d) PMCP 80x80-grid. (e) PMCP 100x100-grid. (f) PMCP 200x200-grid.

Figure 4.25. Crossed Roadside Units Normalized to Standard Gaussian: Grid Setups 80x80, 100x100, and 200x200.



(a) MCP 300x300-grid. (b) MCP 400x400-grid. (c) MCP 800x800-grid.



(d) PMCP 300x300-grid. (e) PMCP 400x400-grid. (f) PMCP 800x800-grid.

Figure 4.26. Crossed Roadside Units Normalized to Standard Gaussian: Grid Setups 300x300, 400x400, and 800x800.

ID, while the y-axis indicates the percentage of the roadside units crossed according to the total of roadside units deployed during the 10 000s of simulation. Thus, every point is computed as indicated in Equation 4.8:

$$P_{\nu,g} = \frac{R_{\nu,g}}{\alpha_g} \tag{4.8}$$

Where:

$P_{\nu,g}$= percentage of roadside units crossed by vehicle ν in grid setup g;

$R_{\nu,g}$= number of roadside units crossed by vehicle ν in grid setup g;
α_g= number of roadside units deployed in grid setup g.

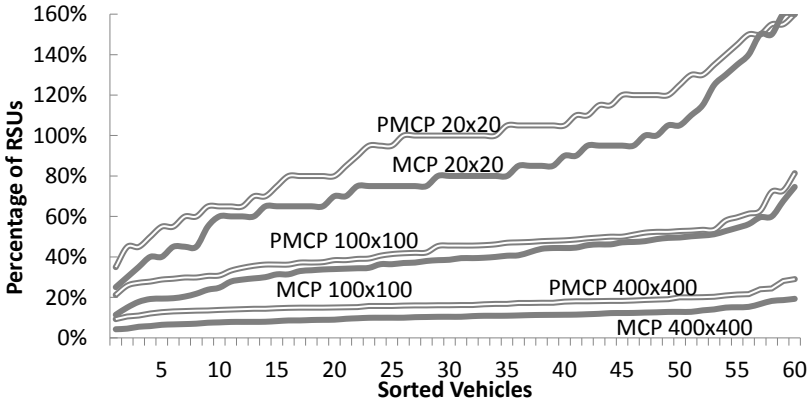


Figure 4.27. Percentage of Crossed Roadside Units for each Vehicle in Grid Setups 20x20, 100x100, 400x400: the x-axis indicates vehicle's ID, while the y-axis indicates the percentage of Roadside Units.

Again, PMCP-based overcomes MCP-based during the entire experiment.

4.7 Analysis

The literature presents several deployment algorithms for roadside units. But, it is not unusual that such algorithms do not receive road networks as input. Such issue happens because a road network is a very complex structure. In this chapter we demonstrated every road structure may be converted onto a grid-based road network through partitioning the road network. When we partition a road network, the complex road network is replaced by a well-behaved and easy-to-understand grid. Vehicles always move between adjacent cells, and this simplifies modeling the vehicles displacements. Partition allows us to generate a very reliable model (arbitrarily) approximated to real road networks. The grid may have its granularity changed according to the application/users need. This self-adapting characteristic makes the grid adequate to real-time routines and to process huge amounts of data iteratively.

We propose the deployment of roadside units as a two-stage process: (i) partition the road network; (ii) apply PMCP to select the α-better cells. We assume each urban cell indivisible. Thus, our results are not affected by the quality of intra-cell deployment. The experiments have characterized selection of urban cells when using PMCP-based and MCP-based approaches. We have presented two sets of experiments: Set S_1 considers the selection of the top-five urban cells in grid setups ranging from

10x10 up to 100x100: our goal is to characterize how PMCP and MCP select the most important cells.

In Experiments S_2 we always cover the same portion of the road network. We cover 5% of the entire road network based on the results reported in Sections 3.5.1 and 4.5.2: Section 3.5.1 reports that, by covering 6.2% of the road network, PMCP-based achieves approximately 100% of vehicle-to-infrastructure contacts, while MCP-based achieves 88%. Section 4.5.2 reports that, by selecting five cells from a 10x10-grid, PMCP-based achieves 99% of contacts, while MCP-based achieves 93%.

PMCP-based always overcomes MCP-based: improvements are due to the fact that PMCP-based considers a mobility model based on turning ratios, leading to a more complete evaluation of the traffic scenario.

4.8 Overview of the Next Chapter

In the next chapter we analyze a realistic scenario composed of a real road network of the Cologne's city, and a realistic mobility trace.

Chapter 5

Realistic Flow Deployment

In Chapter 3 our goal was designing a proposal able to improve the performance of the Maximum Coverage Problem. A real implementation of Maximum Coverage Problem requires previous knowledge of vehicles trajectories. When we lack the trajectories information, the performance of MCP is strongly affected. Thus, we propose a new deployment algorithm (PMCP-based), and we have evaluated the performance of PMCP-based in a theoretical road network with unrealistic traffic. In Chapter 4 we have proposed the partition of the road networks in order to adapt any possible road network to a grid-based road network. We validated the performance of PMCP-based using a real road network with randomized traffic, performing a worst-case analysis of PMCP-based.

Here we perform an average case analysis of our deployment considering the vehicular mobility trace of the city of Cologne, Germany[1] (partially shown in Figure 5.1). PMCP-based divides the road network in a set of adjacent same size cells. We apply PMCP-based (Partitioned Probabilistic Maximum Coverage Problem) and MCP (Maximum Coverage Problem, our baseline) to select most promising partitions and we compare the performance of each approach through several metrics discussed in Section 5.3. Figure 5.1 presents a partial view of the Cologne's road network. As we can notice, it is a very complex road network, and the partitioning plays an important role in reducing such complexity.

We model the deployment of roadside units as a Probabilistic Maximum Coverage Problem, and we propose PMCP-b, a deployment algorithm based on an approximate solution for Probabilistic Maximum Coverage Problem. We compare our deployment algorithm to MCP-g and MCP-based, both heuristics proposed in (Trullols et al., 2010). MCP-g is the greedy solution for Maximum Coverage Problem (Cormen et al., 2001),

[1] `http://kolntrace.project.citi-lab.fr/`

Figure 5.1. Cologne's Road Network.

and it relies on the trajectories information of each vehicle. We say that MCP-g relies on full mobility information. On the other hand, MCP-based does not assume any mobility information at all. Results demonstrate:

- By covering[2] 2.5% of Cologne[3] (selection of 250 urban cells in a set of 10 000), we achieve the following coverage of distinct vehicles: MCP-g=98.1%; PMCP-b=89.6%; MCP-kp=82.8%. PMCP-b improves MCP-kp in 6.8%, while MCP-g improves PMCP-b in 8.5%. We evaluate whether the 8.5% improvement worthies tracking the trajectories of individual vehicles;

- The number of V2I contact opportunities per roadside unit in PMCP-b presents a clear lower bound offering minimum guarantees of efficiency for network designers. Such guarantees are not provided by MCP-kp, and they are too weak in MCP-g;

- A typical vehicle driving in MCP-g experiences at most 25 infrastructure contacts. In PMCP-b, a vehicle experiences up to 60 contacts. In MCP-kp it may reach up to 75 contacts during a single trip. PMCP-b and MCP-kp presents more V2I

[2]We are assuming a slightly different concept of "coverage": traditional usage of coverage indicates a continuous region where users have high probability of connection. But, because we are assuming **infostations** Frenkiel et al. (2000), we consider small islands of coverage: fragmented and possibly disconnect areas where users are supposed to meet connection.

[3]Available traces in: http://kolntrace.project.citi-lab.fr/

contact opportunities because they concentrate the roadside units at popular locations. The strategy to concentrate the roadside units in popular locations also has a side-effect: 17.03% of the MCP-kp vehicles never cross any roadside unit. For a matter of comparison, PMCP-b measure is 10.12%, while MCP-g measure is just 1.86%.

The main contribution of this chapter is the proposal of a deployment algorithm based on a novel paradigm of mobility information, i.e., partial mobility information. Initial deployment works typically propose the allocation of roadside units at the most crowded locations of the road network, and they do not assume any mobility information at all (assumption inherited from the cellular networks).

On the other hand, modern deployment works propose strategies assuming full knowledge of vehicles trajectories (full mobility information). However, full knowledge of vehicles trajectories is a hard assumption when we consider a real deployment because: (i) knowledge of vehicles trajectories implies in several privacy issues; (ii) processing the vehicles trajectories requires a large processing effort: thus, we may not be able to propose dynamic deployment strategies of mobile roadside units; (iii) the network designer may not have the trajectories information available.

Thus, in this chapter we propose a deployment strategy based on partial mobility information, and we compare our approach to a deployment strategy that allocates the roadside units at the most crowded locations (MCP-kp), and a deployment strategy considering full mobility information (MCP-g).

Chapter is organized as follows: Section 5.1 reviews the deployment algorithms under consideration. Section 5.2 presents a comparison of the PMCP-based and MCP-based approaches. Section 5.3 details the experiments considering the realistic scenario. Section 5.4 reevaluates the partitioning of road networks, now considering the Cologne case. Section 5.5 presents an Integer Linear Programming Formulation for the deployment. Section 5.6 presents FPF, a deployment strategy that extends the Projection of the Flow to the entire road network. Section 5.7 concludes the chapter.

5.1 Review of Deployment Algorithms

Trullols et al. (2010) model the allocation of roadside units as a Maximum Coverage Problem. In order to cover a given region, MCP-g iteratively selects those α intersections presenting the largest number of uncovered vehicles ('uncovered' means that a vehicle has not reached any roadside unit yet). Although Maximum Coverage Problem is NP-Hard (Cormen et al., 2001), it is well-known that the greedy heuristic

achieves an approximation factor $(1 - 1/m)^m$, where m is the maximum cardinality of the sets in the optimization domain (Trullols et al., 2010). However, the greedy heuristic requires previous knowledge of vehicles trajectories. Algorithm 1 presents MCP-g.

The heuristic MCP-g chooses sets (i.e., locations) according to one rule: at each stage, choose a set which contains the largest number of uncovered elements. MCP-g requires previous knowledge of vehicles trajectories (full mobility information). Trullols et al. (2010) present an in-depth discussion of MCP-g.

Tracking individual vehicles raises several privacy (and practical) issues, which have motivated us to perform the investigation presented in this work. Notice that when we remove the trajectories information, MCP-g has no means to identify the locations accounting for the maximum number of uncovered vehicles. Without the trajectories information, our only viable choice is to select the most crowded locations to receive the roadside units. Trullols et al. name this heuristic as 0-1 Knapsack Problem (kp). We use MCP-kp as an approximate solution to Maximum Coverage Problem when we do not have access to vehicles trajectories.

MCP-based receives the matrix of locations (M) where each $M_{i,j}$ stores the number of vehicles that have crossed location (i, j), and the number of available roadside units (α) to be deployed. MCP-based selects (at each iteration) the location accounting for maximum number of vehicles . Notice that MCP-based does not use any mobility information. In order to improve the results of MCP-based, we propose modeling the deployment as a Probabilistic Maximum Coverage Problem.

Probabilistic Maximum Coverage Problem is a generalization of the Maximum Coverage Problem: it assumes that each element (vehicle) has a probability p of belonging to a given set (location). Suppose we have (let's say) 100 vehicles at location l_1. We observe that 20 vehicles migrate from l_1 to l_2. So, migration ratio from l_1 to l_2 is 0.2. Thus, $P(l_1, l_2) = 0.2$. P is a stochastic matrix responsible for modeling the mobility of vehicles, and it represents the core of our proposed approach. Notice that tracking individual vehicles is not necessary to infer P.

In this chapter we characterize the efficiency of using partial mobility information by comparing our approach to MCP-g and MCP-based. Because MCP-g relies on full knowledge of vehicles trajectories, MCP-g is supposed to provide a better deployment of roadside units than our strategy. However, tracking individual vehicles seems not to be a realistic assumption in real deployments. Besides that, the selection of a deployment algorithm must take into account several other issues, such as computational cost: low-computational-cost strategies may support **dynamic deployment** and **real time deployment**.

Although dynamic deployment is not possible with stationary infrastructure, several works (Tonguz and Viriyasitavat, 2013; Jerbi et al., 2008; Luo et al., 2010; Annese et al., 2011; Mishra et al., 2011; Sommer et al., 2013) propose the adoption of a virtual-and-mobile infrastructure to support the vehicular communication. Going one step further, we may also consider the deployment as a metaphor to a wider class of resources allocation problem, and we can easily find applications for **real-time deployment**, such as: (i) arrangement of resources for maximum efficiency in the network; (ii) management of content distribution (music, video, games); (iii) caching mechanisms (transferring contents in the right measure to vehicles during the trip), among several others. Algorithm 3 presents our proposed algorithm.

PMCP-based iteratively selects the locations presenting higher projections of flow. When PMCP-b selects a location, it re-computes the number of distinct-and-uncovered vehicles at all remaining locations using migration ratios. As we have demonstrated, PMCP-based increases the quality of deployment when compared to MCP-based. As PMCP-based projects the flow of vehicles, it is able to position roadside units in accordance with mobility. PMCP-based also optimizes number of required roadside units to reach a given share of vehicles (because PMCP-based attenuates vehicles that will be covered in near future).

Projection of the flow has the purpose of discovering the number of vehicles that will reach a given location. Basically, we count the number of vehicles at each one of the preceding locations, and we obtain the projected number of vehicles using stochastic matrix (migration ratios) P. Process is simple: suppose that we have just inserted a new roadside unit within the road network. After inserting the new roadside unit, we use the flow projection to infer the expected number vehicles arriving from all locations that will cross this new deployed roadside unit. Identified vehicles are removed from the model because they will reach this new added roadside unit soon.

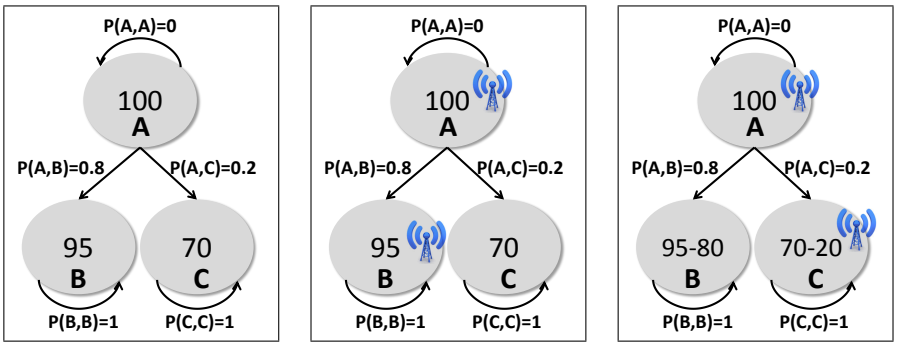
Notice that we can easily adapt our considered deployment algorithms to use a partitioned road network, as shown in Algorithm 8.

Algorithm 8 Adapting Deployment Algorithms to Partitions.

Input: P, ϱ, T, α, ψ, s;
Output: Γ (cells receiving RSUs);
1: $\{M, P, G\}$=Partition(ϱ, T, ψ);
2: **if** (s is mcp-based) **then return** MCP-based(M, α);
3: **else if** (s is pmcp-based) **then return** PMCP-based(M, P, α);
4: **else if** (s is mcp-g) **then return** MCP-g(G, α);
5: **end if**

5.2 Comparison between PMCP-b and MCP-kp

In order to highlight the distinction of MCP-kp and PMCP-b, we present the scenario of Figure 5.2. A given road network is divided into three locations (A, B, C). Inside each location we indicate the number of vehicles crossing such location during the observation period. We also indicate the vehicles migration ratios. For instance, given that a vehicle is in A at time t, it has 80% probability of reaching B at time $t + 1$. Thus, P(A,B)=0.8.



(a) Scenario. (b) MCP-kp selection. (c) PMCP-b selection.

Figure 5.2. Selection of locations to receive the roadside units: (a) Scenario; (b) MCP-kp selection; (c) PMCP-b selection.

Our goal is to select two locations to receive the roadside units. MCP-kp selects the locations that present the highest concentration of vehicles, i.e., locations {A,B}. On the other hand, PMCP-b selects the locations {A, C}. Initially, PMCP-b selects the location presenting the highest traffic (A). Then, it projects the flow. From the 95 vehicles recorded in B, 80 vehicles are already covered in A. Thus, selecting B implies in covering just another 15 vehicles. Similarly, from the 70 vehicles recorded in C, 20 vehicles arrive from A. Thus, C implies in covering 50 new vehicles, and then PMCP-b selects {A,C}.

5.2.1 Evaluation Over a Synthetic Road Network

Now we present a first comparison among PMCP-b, MCP-kp, and MCP-g considering a synthetic 9x9 grid road network where vehicles always move bottom-up and left-right. Here we assume that roads intersections define "locations", and we compute the projection of the flow heading to the intersections. Thus: **In this section we consider that the stochastic matrix P holds turning ratios at roads intersections**. Figure 5.3 illustrates our road network.

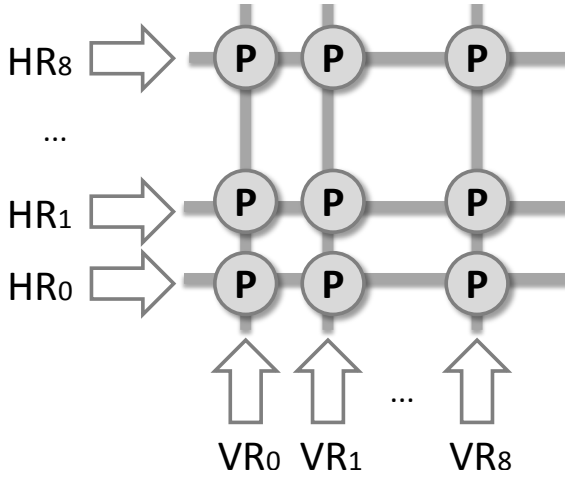


Figure 5.3. 9x9 Grid Road Network.

Obviously, this is a quite unrealistic scenario, and the purpose of this set of experiments is just to illustrate how our strategy works. We built a simulator emulating the grid road network. Our simulator inserts vehicles at the bottom and left borders of the grid. When vehicles reach the top or right borders, they leave the simulation. Our simulator dispatches one vehicle at a time in a random walk trip inside the grid. Random walk is guided by the stochastic matrix P (turning ratios at roads intersections). Simulator works as follows:

5.2.1.1 Synthetic Road Network Simulation Algorithm

In **step 1** we get a random number to define where the vehicle will start the trip. There are 18 possible start points: 9 start points in the bottom borders ($VR_0..VR_8$), and 9 start points in the left borders ($HR_0..HR_8$). In **step 2**, the vehicle is in the grid and the trip starts. While the vehicle is inside the grid, we loop: we get a random number which determines whether the vehicle has to move up or right. The vehicle always moves between adjacent cells. Whenever a vehicle reaches a cell, we increment the "counter of visits" for that cell. When the vehicle leaves the grid, we return to step 1. We repeat this process β times, where β represents the number of vehicles that we intend to simulate. Then, we apply PMCP-b, MCP-kp, and MCP-g to locate the roadside units.

Below we present a detailed case study considering a scenario where the **turning ratio is 0.5 for all intersections** composing our road network (recall that the two possible movement directions: bottom-up, and left-right). During the simulation **we have dispatched β=10 000** vehicles.

5.2.2 Selection of Locations made by PMCP-b and MCP-kp

Let's turn our attention to the selection of locations made by PMCP-b and MCP-kp. Figure 5.4 presents the number of vehicles crossing each intersection (we also present a heatmap where the darker the region, the more intense is the flow). We notice a very small variation from one intersection to another. Such small variation occurs because we have executed the simulation only once (we highlight that such variation is zero if we launch infinite vehicles). The selected intersections for seven roadside units made for PMCP-b and MCP-kp are presented below:

	VR_0	VR_1	VR_2	VR_3	VR_4	VR_5	VR_6	VR_7	VR_8
HR_8	1150	1134	1098	1047	1115	1085	1094	1071	1130
HR_7	1139	1102	1051	1043	1090	1080	1127	1094	1127
HR_6	1092	1070	1028	1088	1095	1108	1124	1091	1134
HR_5	1079	1064	1075	1087	1073	1119	1123	1120	1163
HR_4	1077	1053	1052	1086	1136	1155	1165	1099	1117
HR_3	1007	1058	1081	1145	1128	1116	1122	1046	1133
HR_2	1096	1138	1154	1196	1147	1150	1090	1034	1131
HR_1	1107	1096	1131	1120	1064	1098	1048	1119	1198
HR_0	1141	1123	1117	1088	1102	1137	1081	1139	1172

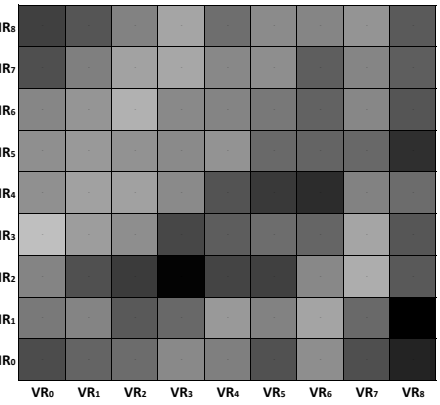


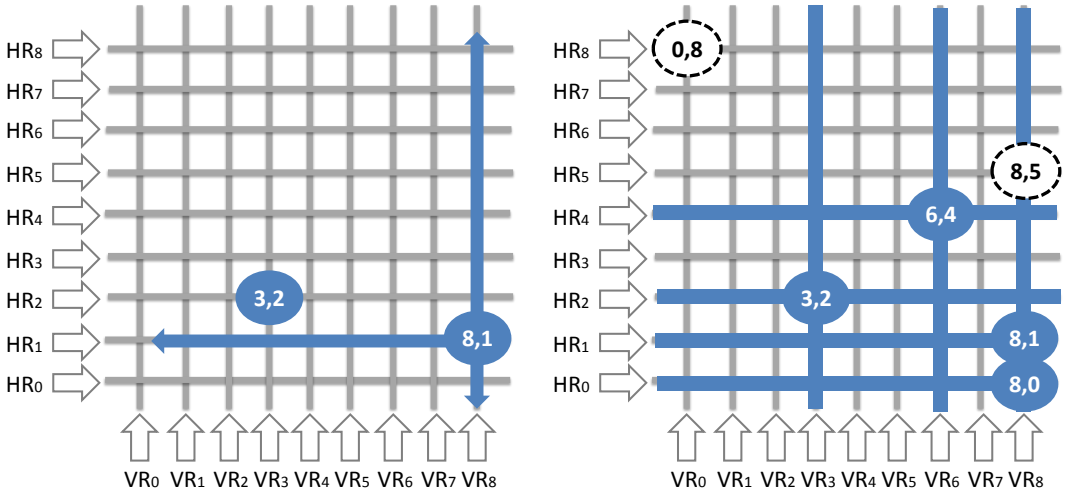
Figure 5.4. Number of Vehicles Crossing the Intersections.

MCP-kp={{8,1},{3,2},{8,0},{6,4},{8,5},{5,4},{2,2}};

PMCP-b={{8,1},{3,2},{8,0},{6,4},{0,8},{5,2},{0,0}};

The first intersection selected by both heuristics is the one presenting higher traffic. For the remaining intersections, MCP-kp continues to select intersections based on the level of traffic, while PMCP-b selects those intersections presenting the highest projections of the flow. When we look at the set of selected intersections for MCP-kp and PMCP-b, we notice that both heuristics have selected the same four intersections {{8,1},{3,2},{8,0},{6,4}}. Let's analyze this selection in details:

The intersection presenting the highest traffic is {8,1}, which is selected by both heuristics. The second intersection presenting the highest traffic is {3,2}, thus it is selected by MCP-kp. However, such intersection is also selected by PMCP-b because the projection of the flow at {3,2} is not affected by the already deployed roadside unit at {8,1}, as illustrated in Figure 5.5(a). The same reasoning applies for roadside units {8,0} and {6,4}.



(a) RSUs in {8,1} and {3,2}. (b) PMCP-b replaces {8,5} per {0,8}.

Figure 5.5. Analysis of the Deployment Scenario.

The selection of the fifth location is distinct for PMCP-b and MCP-kp. While MCP-kp selects intersection {8,5}, PMCP-b selects {0,8} because the projection of the flow from intersections {8,0} and {8,1} reduces the number of uncovered vehicles at {8,5}. Figure 5.5(b) illustrates this. Also notice that the roadside unit deployed at {8,1} does not reduce the number of uncovered vehicles at {8,0} because the flow moves bottom-up and left-right. Thus, no vehicle moves from {8,1} to {8,0}, and inserting a roadside unit at {8,1} does not change the number of uncovered vehicles at {8,0}.

5.2.3 Coverage Evaluation

Figure 5.6 presents the number of distinct vehicles experiencing at least one V2I contact for a given number of deployed roadside units. The x-axis indicates the percentage of distinct vehicles presenting contact opportunities. The y-axis indicates the number of deployed roadside units (from 1 up to 7). Obviously, MCP-g overcomes both PMCP-b and MCP-kp, after all, MCP-g considers the full trajectories information for all vehicles, and thus it is able to select better locations for deploying the roadside units.

PMCP-b overcomes MCP-kp after deploying the fifth roadside unit (illustrated in Experiment 5.6.2.2). In practice, when we deploy the seventh roadside unit, PMCP-b presents an improvement of 20% over MCP-kp.

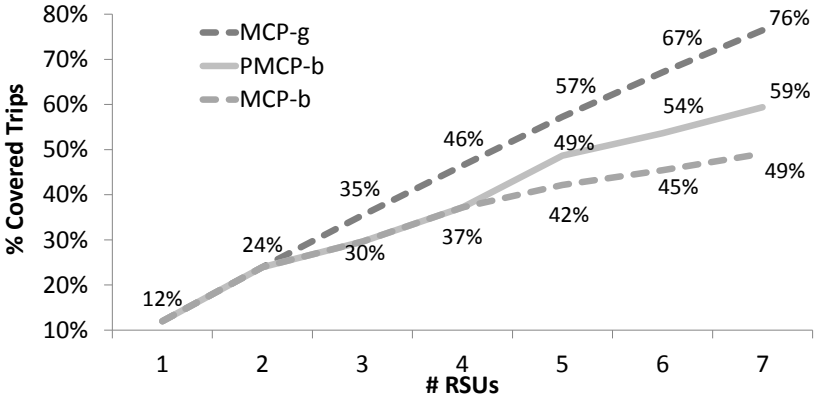


Figure 5.6. Coverage Evaluation. The x-axis indicates the number of distinct vehicles experiencing at least one V2I contact opportunity. The y-axis indicates the number of deployed roadside units.

5.2.4 Redundant Coverage

Now we evaluate the number of vehicles experiencing two or more V2I contact opportunities in order to infer the level of redundant coverage. In Figure 5.7, the x-axis indicates the roadside units ID (the ID indicates the order of deployment), while the y-axis indicates the number of vehicles crossing more than one roadside unit. MCP-g only presents redundant coverage after the sixth roadside unit.

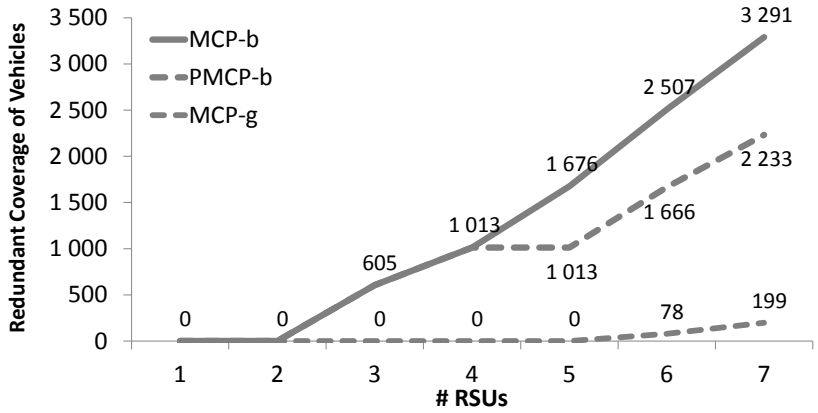


Figure 5.7. Redundant Coverage. The x-axis indicates the roadside units ID (order of deployment), while the y-axis indicates the number of vehicles crossing more than one RSU.

PMCP-b and MCP-kp present redundant coverage when deploying the third roadside unit. Notice that the fifth roadside unit deployed by PMCP-b does not present redundant coverage, an effect of the projection of the flow. The sixth roadside unit deployed by PMCP-b is {5,2} and it results in a redundant coverage of 653 vehicles.

On the other hand, the sixth roadside unit deployed by MCP-kp results in a redundant coverage of 831 vehicles. An inspection of Figure 5.5(b) indicates that the number of uncovered vehicles at {5,2} is reduced because of the projection of the flow coming from {3,2}. After deploying seven roadside units, PMCP-b reduces in 32% the number of vehicles presenting redundant coverage (from 3 291 to 2 233).

5.2.5 Discussion

So far we have introduced the Probabilistic Maximum Coverage Problem and we have proposed the PMCP-b (Algorithm 3) greedy heuristic that approximates the solution of PMCP when we do not have access to the vehicles trajectories. We compare PMCP-b to two baselines:

- MCP-g is the greedy solution for the Maximum Coverage Problem. It requires full knowledge of vehicles trajectories. Thus, MCP-g relies on full mobility information. On the other hand:

- MCP-kp locates the roadside units at locations presenting the highest traffic. Thus, MCP-kp does not rely on any mobility information at all.

In the first iteration, PMCP-b selects the location presenting the highest traffic. Every time PMCP-b selects a location, we use the projection of the flow to remove vehicles that will drive through this recently added roadside unit. The remainder selection of locations consider just the projection of the flow. The most important assumption of our work is we do not have access to vehicles trajectories. In this section we have presented a simple deployment scenario with didactic purposes. Although such road network is very far from a real scenario, these experiments demonstrate the basic principles that will guide our investigation in the following sections.

5.3 Comparison of PMCP-based, MCP-based and MCP-g

Now we present a set of experiments comparing the performance of PMCP-b, MCP-kp, and MCP-g considering the city of Cologne, Germany. Our goal is to quantify the impact of the mobility information (none, partial, and full) on their performance. Recall that MCP-kp lacks any mobility information: it just sorts the urban cells and selects those α urban cells presenting the highest volume of vehicles.

On the other hand, PMCP-b considers partial mobility information (migration ratios between adjacent urban cells): it starts by picking the most crowded urban cell. Then, it projects the flow of vehicles according to the stochastic matrix of migration ratios (P), and it selects the urban cell presenting the highest expectancy of vehicles. Then, it projects the flow once again, and selects another urban cell. PMCP-b repeats this process α times. Finally, MCP-g considers full mobility information (trajectories of vehicles): it always selects the urban cells containing the highest number of uncovered vehicles. All experiments are performed using the SUMO[4] simulator and a set of tools designed by our team.

SUMO runs the Cologne scenario and outputs the location of each vehicle (our mobility trace T). Algorithm Partition reads the mobility trace, computes the bounding box of Cologne, partitions Cologne into a grid of $\psi \times \psi$ urban cells, and then translates the mobility trace from Cartesian coordinates to Grid coordinates. After that, we run the Algorithms MCP-g, MCP-kp, and PMCP-b: each of these algorithms outputs an individual file containing the position of the roadside units (in urban cells). Then, we run the Algorithm MeasureFlow: it receives as input the mobility trace and a file containing the location of the α roadside units, and outputs several measures used to evaluate the deployment performance.

We have validated the Program Flow by generating an Integer Linear Programming Formulation Dantzig (1998) of MCP-g into IBM CPLEX Optimizer[5]. IBM CPLEX outputs the location of the roadside units and the status of all vehicles (covered or non-covered). When we run the Algorithm MeasureFlow using the IBM CPLEX output we achieve the exact same results. Algorithms MCP-g, MCP-kp, and PMCP-b were validated through the inspection of small scenarios (such as the scenario presented in Section 5.2.1).

5.3.1 Covered Vehicles x Covered Area

Figure 5.34 presents the percentage of vehicles experiencing at least one V2I contact opportunity when we deploy the roadside units according to PMCP-b, MCP-kp, or MCP-g. The x-axis shows the percentage of covered area (and the number of deployed roadside units), while the y-axis indicates the percentage of covered vehicles.

By covering 2.5% of Cologne (selection of 250 urban cells in a set of 10 000), we reach the following percentage of vehicles: MCP-g=98.1%; PMCP-b=89.6%; MCP-kp=82.8%. PMCP-b improves MCP-kp in 6.8%, while MCP-g improves PMCP-b

[4]Sumo Simulator: `http://sumo-sim.org`.

[5]IBM CPLEX Optimizer: `http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/`.

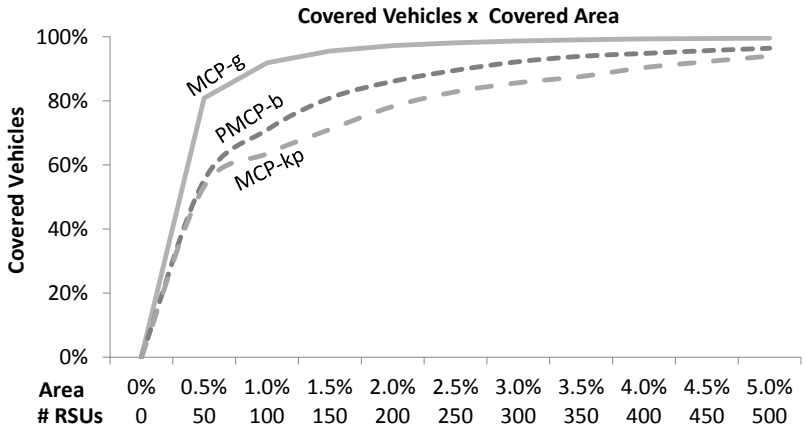


Figure 5.8. Percentage of Covered Vehicles x Covered Area. The x-axis shows the number of deployed roadside units. The y-axis indicates the percentage of vehicles experiencing at least one V2I contact opportunity.

in 8.5%. We must evaluate whether the 8.5% improvement worthies tracking the trajectories of vehicles. When we increase the covered area in 100%, we improve MCP-g in +1.5%, PMCP-b in +6.9%, and MCP-kp in +11.2%. Table 5.8 summarizes these results.

Table 5.1. Covered Area x Covered Vehicles.

	Scenario #1	Scenario #2	Perc. Improvement
Covered Area	2.5%	5.0%	100%
Deployed Roadside Units	250	500	100%
MCP-g	98.1%	99.6%	+1.5%
PMCP-b	89.6%	96.5%	+6.9%
MCP-kp	82.8%	94.0%	+11.2%

Now we analyze the improvements achieved when using a more detailed mobility information. Figure 5.35 plots the improvements of PMCP-b over MCP-kp (Relation i_1), and the improvements of MCP-g over PMCP-b (Relation i_2). The relation i_1 is given by $\frac{PMCP-b}{MCP-kp}-1$, and it peaks at 14% when x-axis=1.5%. Relation i_2 is $\frac{MCP-g}{PMCP-b}-1$, and it peaks at 42% when x-axis=0.5%. When x-axis is 2.5% (250 RSUs), i_1 is 10% and i_2 is 8%. When x-axis is 5% (500 RSUs), both improvements decrease to approximately 3%, indicating a probable saturation of roadside units.

5.3.2 Low Scale Deployment

Figure 5.36 presents an incremental view of the deployment considering small increments of 10 newly deployed roadside units.

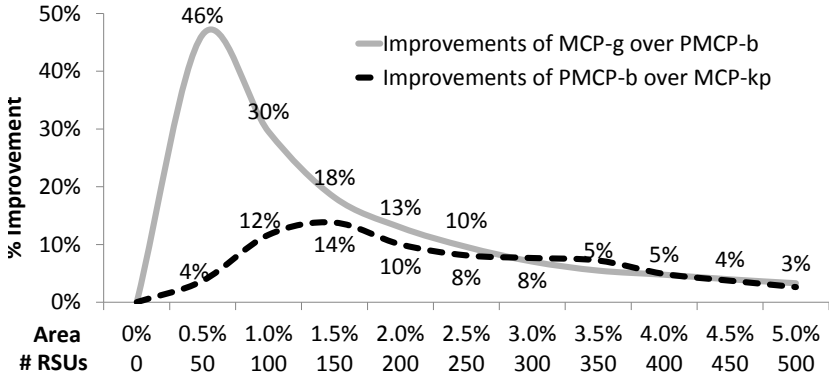


Figure 5.9. Improvements of MCP-g over PMCP-b, and PMCP-b over MCP-kp in terms of the trips experiencing at least one V2I contact opportunity. The x-axis indicates the number of roadside units deployed. The y-axis indicates the percentage improvement.

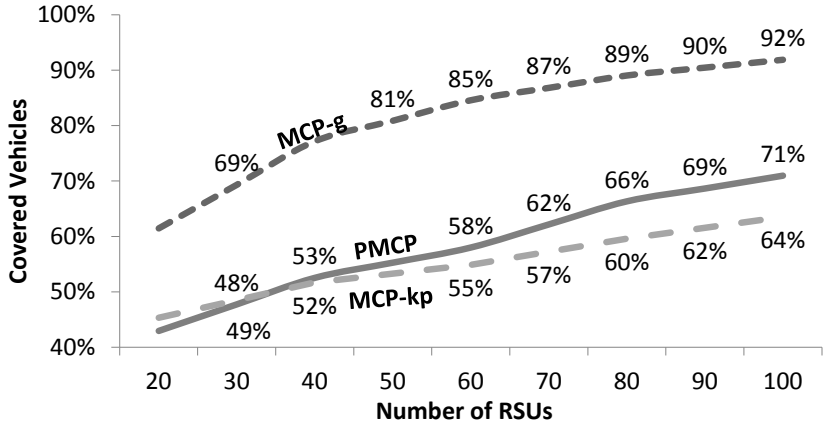


Figure 5.10. Low Scale Deployment. The x-axis indicates the number of deployed roadside units. The y-axis indicates the percentage of vehicles experiencing at least one V2I contact opportunity.

PMCP-b only overcomes MCP-kp by 5% after the deployment of 70 roadside units. In general terms, MCP-g improves by 50% approximately the performance of MCP-kp and PMCP-b.

5.3.3 Vehicles Reaching the Infrastructure Over Time

Here we evaluate the performance of the roadside units over time by considering the deployment of 250 roadside units. Figure 5.11 plots the number of distinct vehicles contacting the infrastructure at least once over time. The x-axis indicates time (×1 000s), and y-axis indicates the percentage of vehicles experiencing at least one contact.

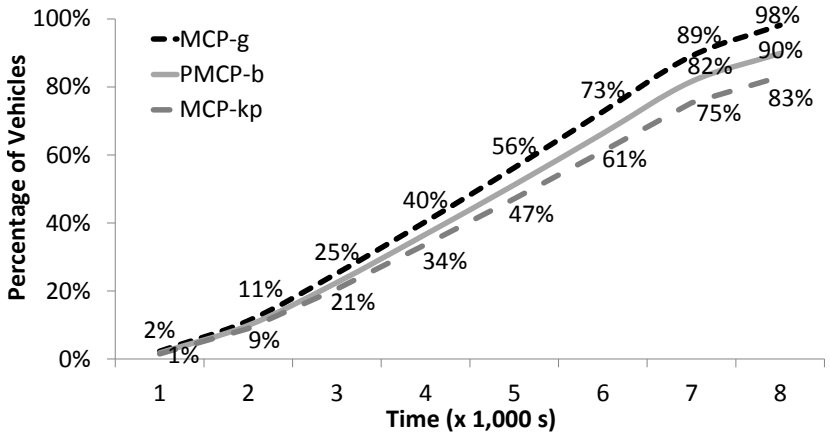


Figure 5.11. Percentage of vehicles reaching the infrastructure over time. The x-axis indicates the time (×1 000s). The y-axis indicates the number of vehicles.

The experiment shows that after 8 000s of simulation over 90% of all vehicles drove past at least one of the roadside units deployed according to PMCP-b. MCP-g accounts for 98% (+8%), while MCP-kp achieves 83% (-7%). Thus, the partial mobility information used by PMCP-b seems an interesting trade-off between drivers privacy and deployment quality.

5.3.4 Number of V2I Contacts per Roadside Unit

In Sections 5.3.1 and 5.3.2 we have characterized the number of distinct vehicles experiencing at least one V2I contact opportunity. Now we characterize the total number of V2I contact opportunities (one vehicle may contact the infrastructure more than once during the trip).

Figure 5.37 plots the number of V2I contact opportunities per roadside unit considering the MCP-kp deployment. The x-axis indicates the ID of the roadside unit (sequentially from 1 to 250 following the order of deployment), while the y-axis indicates the number of V2I contact opportunities. The plot indicates a large variance on the number of contacts, with several peaks and anti-peaks. It means that adding a new roadside unit using MCP-kp is a risky task in terms of the **return on investments** by network providers. As MCP-kp always selects those cells having the highest density of vehicles, it results in high redundant coverage: high traffic areas tend to be adjacent, what explains the "peak-followed-by-anti-peaks" pattern of MCP-kp.

Figure 5.38 presents the same result, but now considering a MCP-g deployment. MCP-g presents (approximately) the same lower bound for every roadside unit. Because MCP-g considers individual vehicular mobility, the contribution of each

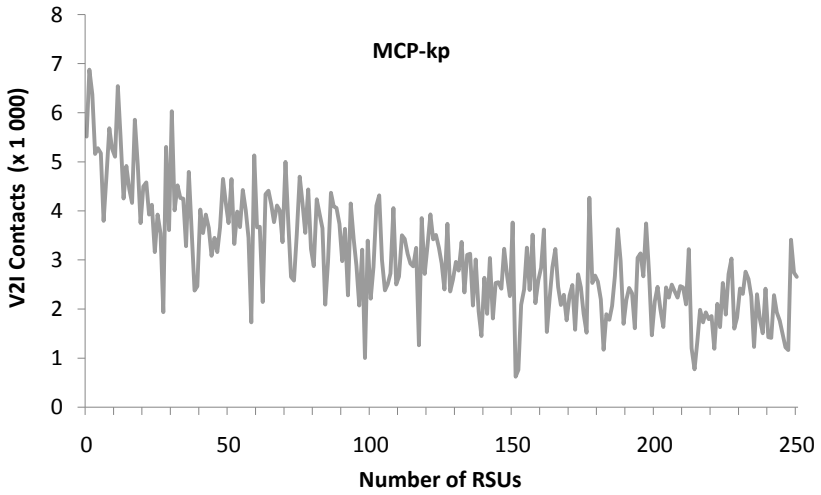


Figure 5.12. MCP-kp: Number of V2I contacts per RSU. The x-axis presents the RSUs in the order of deployment. The y-axis indicates the total number of contacts for each roadside unit.

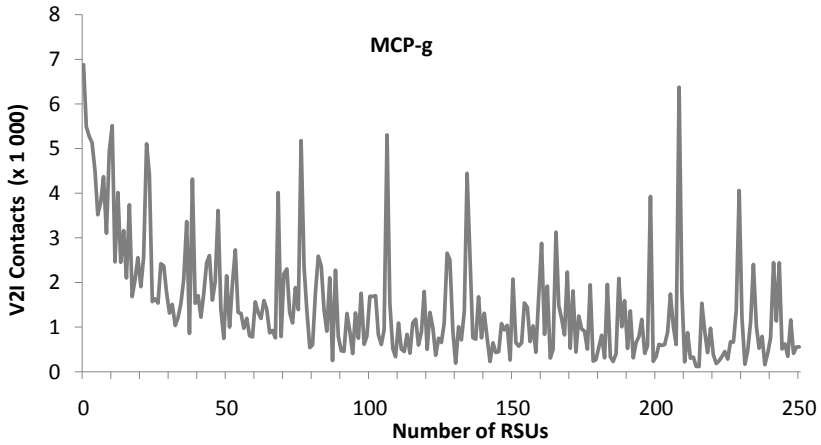


Figure 5.13. MCP-g: Number of V2I contacts per RSU. The x-axis presents the RSUs in the order of deployment; The y-axis indicates the total number of contacts for each roadside unit.

roadside unit appears equally balanced. On the other hand, such equal balance may not provide enough information for a network designer since the lower bound is very low.

Figure 5.39 presents the same result, but now considering a PMCP-b deployment. As we can notice, the efficiency of PMCP-b is much more deterministic. PMCP-b has a clear lower bound offering minimum guarantees of efficiency for network providers. The plot shows peaks, but not depressions. When we consider total number of contacts, MCP-kp achieves 796 021 contacts, followed by PMCP-b with 725 170 (-8.9%) and

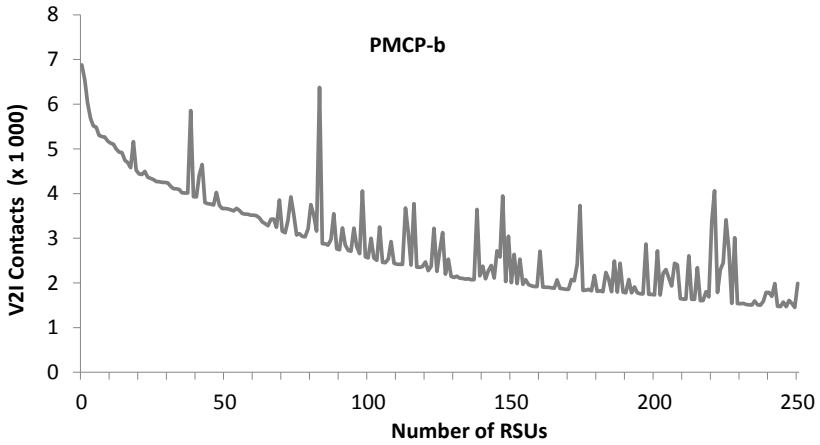


Figure 5.14. PMCP-b: Number of V2I contacts per RSU. The x-axis presents the RSUs in the order of deployment; The y-axis indicates the total number of contacts for each roadside unit.

MCP-g with 370 098 (-53.4%).

5.3.5 V2I Contacts per Vehicle

Now we analyze the distribution of V2I contact opportunities per vehicle. Figure 5.40 presents a histogram indicating the distribution of vehicles per number of crossed roadside units. The x-axis indicates the number of crossed roadside units. The y-axis indicates the number of vehicles crossing that amount of roadside units. We used a logarithmic scale in the y-axis.

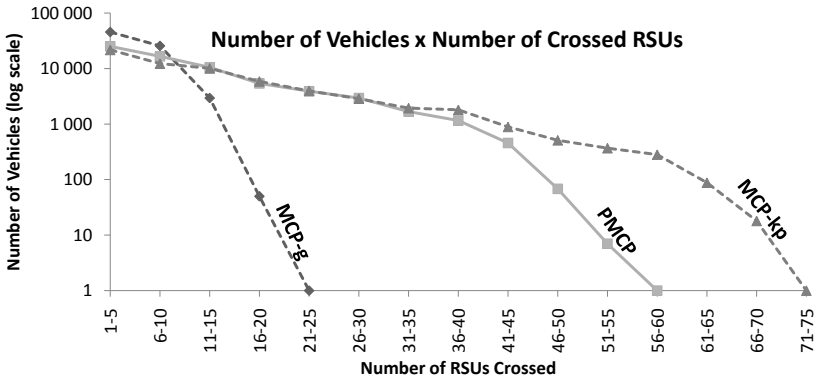


Figure 5.15. Number of crossed RSUs per vehicle. The x-axis indicates the amount of RSUs crossed. The y-axis indicates the number of vehicles (logarithmic scale).

MCP-g presents vehicles reaching up to just 25 roadside units. PMCP-b presents vehicles reaching up to 60 roadside units, while MCP-kp presents vehicles reaching up

to 75 roadside units. MCP-kp presents more vehicles driving through more roadside units because it concentrates roadside units in very popular locations, and vehicles crossing such popular locations experience several contact opportunities.

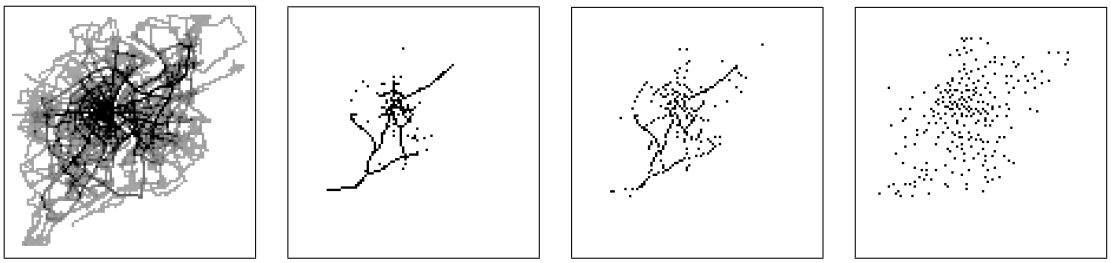
The strategy to concentrate roadside units in very popular locations also has a side-effect: 17.03% of the MCP-kp vehicles never cross any roadside unit. For a matter of comparison, the same measure for PMCP-b is 10.12%, while for MCP-g it is just 1.86%. Thus, PMCP-b is able to reduce the redundant coverage provided by MCP-kp, but MCP-g provides a better distribution of V2I contacts.

5.3.6 Roadside Units Layout

Now we present the layout of the roadside units deployed according to each strategy. The map of the city of Cologne is presented in Figure 5.41(a). The level of black is directly proportional to the level of traffic.

The deployment proposed by MCP-kp is indicated in Figure 5.41(b). Black dots indicate roadside units. Layout of PMCP-b is presented in Figure 5.41(d). When we compare both strategies we notice that PMCP-b changes the location of several several roadside units placed by MCP-kp. Such changes results from the projection of the flow of vehicles, producing a layout that better fits the displacement of vehicles.

Figure 5.41(c) presents the deployment suggested by MCP-g, clearly a more-distributed layout, after all, MCP-g relies on full vehicles trajectories.



(a) Road network. (b) MCP-kp. (c) PMCP-b. (d) MCP-g.

Figure 5.16. Road network and layout of roadside units for MCP-kp, PMCP-b, and MCP-g.

5.3.7 Characterizing Migration Ratios

Here we characterize stochastic matrix (P) of Cologne. Each entry $P_{\phi1,\phi2}$ holds the probability that a vehicle traveling on road $\phi1$ continues on $\phi1$ when $\phi1$ intersects $\phi2$. Each $P_{\phi1,\phi2}$ is, in fact, a vector of four positions because we have to store the probability

of four possible movements: up, down, left, and right. Although we have divided the city of Cologne in 10 000 cells, following analysis consider just cells presenting traffic level > 0. Then, our analysis is restricted to just 4 152 cells. Table 5.2 presents a basic statistical description of P considering the four possible movements. As we expect, stochastic matrix is balanced among the four possible directions, after all it reflects the flow of 75 515 vehicles.

Table 5.2. Statistical Description of P

measure	up	down	left	right
average	0.214120	0.206923	0.195594	0.1900843
std. error	0.004103	0.004051	0.003897	0.0038346
median	0.109482	0.086957	0.082246	0.0778895
std. dev.	0.264411	0.261039	0.251123	0.2470866
variance	0.069913	0.068141	0.063062	0.0610518
curtosis	1.013997	0.712298	1.172820	1.2952059
min	0.000000	0.000000	0.000000	0.0000000
max	1.000000	1.000000	1.000000	1.0000000
count	4 152	4 152	4 152	4 152
confidence (95%)	0.0080449	0.0079424	0.0076406	0.007517

Table 5.3 presents the covariance between entries the four possible directions: up, down, left, and right.

Table 5.3. Covariance of P

	up	down	left	right
up	0.069896352	-	-	-
down	-0.017733371	0.068125262	-	-
left	-0.021451657	-0.018467343	0.063047663	-
right	-0.019032246	-0.019967146	-0.011687705	0.061037125

Figure 5.17 presents the histogram representing the distribution of probabilities in P: x-axis indicates turning probability, while y-axis indicates number of intersections. Four bars indicate probability of a vehicle moving: up, down, left, or right.

5.4 Road Networks Partition: Evaluation On a Realistic Scenario

In Section 5.3 we show experiments characterizing the deployment performance when we change the grid setup. Such experiments consider a randomized vehicles flow. Now

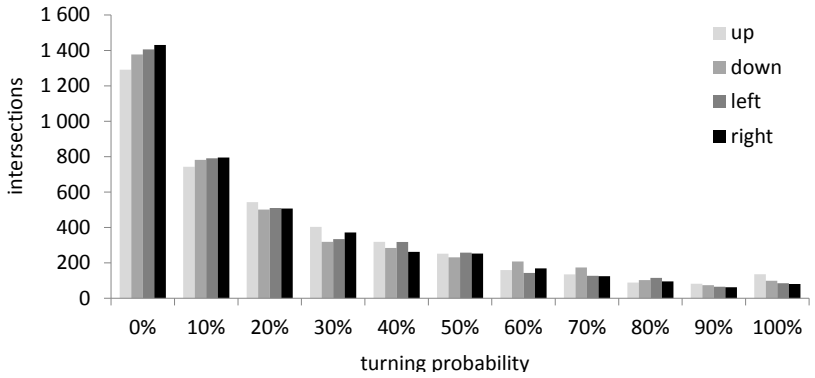


Figure 5.17. Histogram presenting distribution of probabilities in P: x-axis indicates turning probability, while y-axis indicates number of intersections. Four bars indicate the probability of a vehicle moves: up, down, left, or right.

we revisit such experiments, but now we consider a realistic flow. Here we study the partition technique considering MCP-g presented in Algorithm 1. Recall that MCP-g is the greedy solution for the Maximum Coverage Problem. Thus, it relies on previous knowledge of vehicles trajectories. Our selection of MCP-g is based on the fact that MCP-g achieves close-to-optimal coverage (result presented in Chapter 5.5). We propose the deployment of roadside units covering just 0.5% of the road network. Thus, after partitioning the road network we select the 0.5% of the most promising urban cells to receive the roadside units (in Chapter 4 we consider 5% of urban cells receiving roadside units). Table 5.4 highlights the most important distinction between the studies presented in Chapter 4 and the studies presented now.

Table 5.4. Deployment Features

	Chapter 4	Chapter 5.4
Mobility Trace	Randomized	Realistic
Deployment Strategies	PMCP-based, MCP-based	MCP-g
Considers Individual Trajectories	No	Yes
Covered Area	5%	0.5%

5.4.1 V2I Contact Opportunities

Figure 5.18 presents the V2I contact opportunities when we partition the road network into distinct grid setups ranging from 20x20 up to 660x660. We notice a clear convergence of the deployment performance towards the maximum achievable V2I contact opportunities. Figure 5.19 highlights the V2I contact opportunities for grid setups between 140x140 and 660x660. Grid setup 140x140 offers 98.0% of V2I contact

opportunities, while maximum achievable performance is 99.8% (firstly achieved in grid setup 380x380). In both plots, the x-axis indicates the grid setup, while the y-axis indicates the percentage of vehicles presenting at least one contact opportunity.

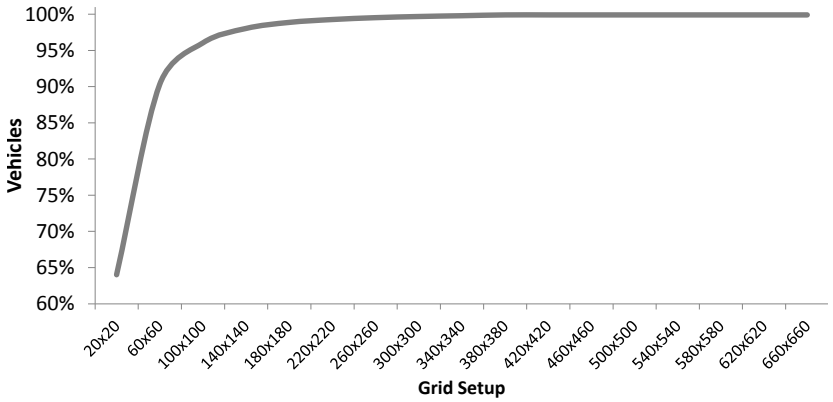


Figure 5.18. V2I Contact Opportunities: Figure presents the V2I contact opportunities for several grid setups. 0.5% of the urban cells receive roadside units.

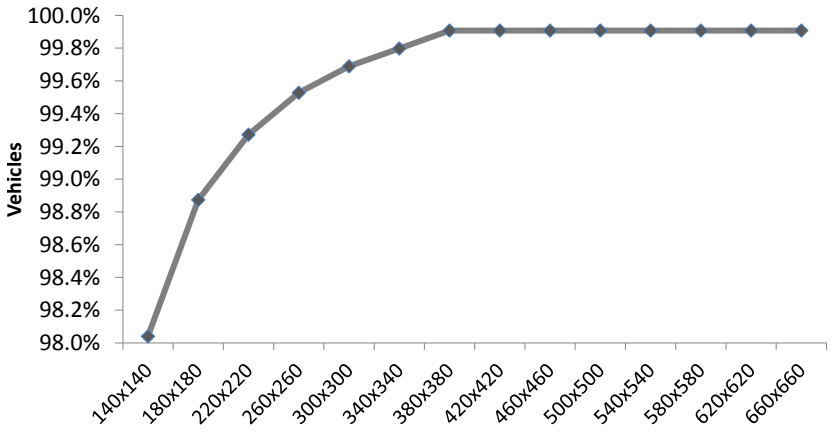


Figure 5.19. V2I Contact Opportunities: Figure highlights presents the V2I contact opportunities for several grid setups. 0.5% of the urban cells receive roadside units.

5.4.2 V2I Contact Time

Figure 5.20 presents the V2I contact time. The x-axis indicates the grid setup, while the y-axis indicates the contact time in millions of seconds (x1 000 000s). The plot peak is also achieved when grid setup is 380x380.

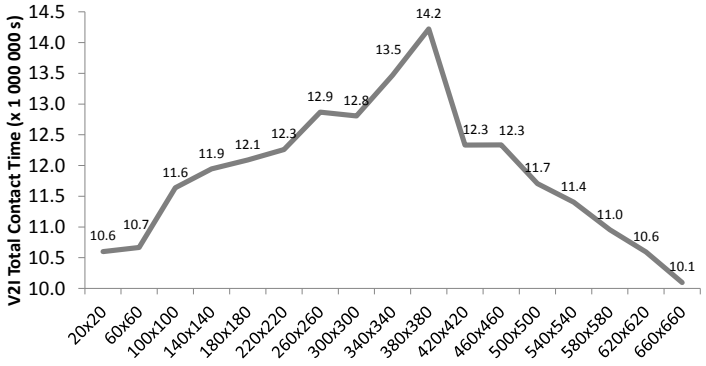


Figure 5.20. V2I Contact Time for Distinct Grid Setups: The x-axis indicates the grid setup, while the y-axis indicates the contact time in millions of seconds (x1 000 000s).

5.4.3 Average V2I Contact Opportunities per Vehicle

Figure 5.21 presents the average number of roadside units crossed per vehicles during the simulation. The x-axis indicates the grid setup, while the y-axis indicates the number of V2I contacts. Again, the peak is when grid setup is 380x380.

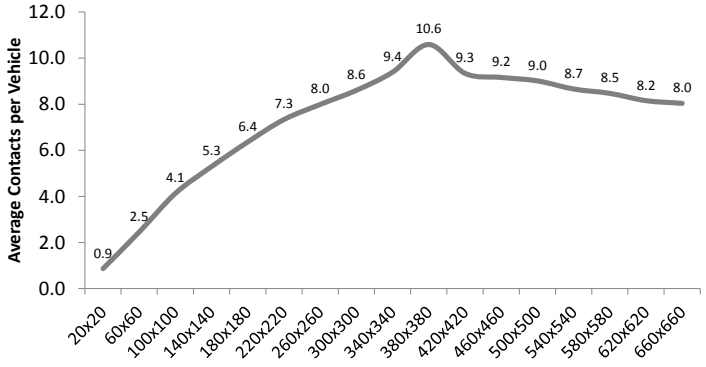


Figure 5.21. Average V2I Contact Opportunities: The x-axis indicates the grid setup, while the y-axis indicates the average number of V2I contacts.

5.4.4 Analysis

In this section we have presented the analysis of partitioning the Cologne's road network. Our analysis considers MCP-g (Algorithm 1) deploying roadside units is 0.5% of all urban cells. Our main findings are:

- Over-partitioning does not increase the deployment performance indefinitely: partitioning the Cologne road network into a 140x140 grid yields 98.0% of V2I

contact opportunities, while partitioning the same scenario in a 660x660 grid yields an increment of just 1.8%. In fact, there is no gain in partitioning the Cologne scenario in more than 380x380 urban cells;

- Contact time is maximized when we partition the Cologne scenario in a 380x380 grid. Beyond such threshold, the contact time decreases. Thus, over-partitioning has a side effect of degenerating some aspects of the deployment;

- Average V2I contact opportunities are maximized when grid setup is 380x380;

Considering these previous results, our best choice is to partition the Cologne scenario in a 380x380 grid setup. In the next section we present an Integer Linear Programming Formulation for the deployment problem targeting the maximization of the number of distinct vehicles experiencing at least one V2I contact opportunity.

5.5 Optimal Deployment

Here we evaluate the performance of our PMCP-based approach compared to the maximum achievable performance. We define an Integer Linear Programming Formulation to solve the optimal deployment with the goal to maximize the number of distinct vehicles experiencing at least one V2I contact. We call this: Opt_{dv}. Our formulation has the following parameters and variables: Parameters of our formulation are: K, α, C, and M. Let K be the set of vehicles, where $K = \{1, 2, ..., k\}$. Let α denotes the number of available roadside units. Let C be the set representing all road network partitions, where $C = \{1, 2, .., c\}$. Let M be the set representing trajectories. Thus:

$$M_{c,k} = \begin{cases} 1, & \text{if vehicle } k \text{ crosses urban cell } c \\ 0, & \text{otherwise.} \end{cases}$$

Variables are A and V: Let A represent urban cells receiving roadside units. Thus:

$$A_c = \begin{cases} 1, & \text{if urban cell } c \text{ receives a roadside unit} \\ 0, & \text{otherwise.} \end{cases}$$

Let V represent 'covered' vehicles. Thus:

$$V_k = \begin{cases} 1, & \text{if vehicle } k \text{ has crossed a roadside unit} \\ 0, & \text{otherwise.} \end{cases}$$

We model the deployment of roadside units as follows:

$$\max \sum_{n=1}^{k} V_n \tag{5.1}$$

Subject to:

$$\sum_{\forall c \in C \mid M_{c,k}=1} A_c \leq \alpha \tag{5.2}$$

$$\sum_{\forall c \in C \mid M_{c,k}=1} A_c \geq V_k \quad \forall k \in K \tag{5.3}$$

$$V_k \geq A_c \quad \forall c \in C, \forall k \in K \mid M_{c,k} = 1 \tag{5.4}$$

$$A_c \in \{0,1\} \quad \forall c \in C \tag{5.5}$$

$$V_k \in \{0,1\} \quad \forall k \in K \tag{5.6}$$

Objective function 5.1 maximizes the number of distinct vehicles reaching roadside units. Constraint 5.2 ensures the number of selected urban cells is $\leq \alpha$. Constraint 5.3 ensures that whenever vehicle k is covered, an urban cell crossed by k has a roadside unit. Constraint 5.4 ensures that whenever vehicle k crosses an having a roadside unit, than the vehicle k is covered. Constraints 5.5 and 5.6 ensures A and V as binary.

The optimization model was executed on IBM CPLEX Optimizer[6]. We compare our PMCP-based approach, MCP-based, and Opt_{dv} through several experiments. Recall that Cologne scenario consists of 75 515 vehicles traveling in a 756km^2 road network during approximately 9 000s. Our experiments reveal that our Integer Linear Programming Formulation is able to solve instances presenting large number of roadside units. Thus, we assume the deployment of α=500 roadside units (coverage of 5% of the entire road network). We were not able to solve the most interesting instances ($10 \leq \alpha \leq 200$). However, when we offer initial solutions computed using the well-know greedy solution for MCP (MCP-g) presented in Algorithm 1, CPLEX returned that optimal solution is at most 3.5% above the MCP-g solution.

This section is organized as follows: Section 5.5.1 compares the number of distinct vehicles contacting the infrastructure. We have segmented the experiment into time intervals. Our goal is to reveal how Opt_{dv} and PMCP-based perform over time. Section 5.5.2 analyzes the number of distinct vehicles crossing each roadside unit. Our goal is to investigate the marginal contribution of each roadside unit. Section 5.5.3

[6] `http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/`

analyzes V2I contact time. Contact time is not part of the objective function, and we intend to reveal how Opt_{dv} performs when compared to a PMCP-based approach. Section 5.5.4 presents the number of roadside units crossed per vehicle. Section 5.5.5 reveals how Opt_{dv} and PMCP-based perform when we vary the grid setup. Section 5.5.6 presents the processing time.

5.5.1 Distinct Vehicles per Time Interval

Here we analyze the number of distinct vehicles experiencing at least one V2I contact. We consider time intervals of 1 000s. Our goal is to reveal how Opt_{dv} and PMCP-based perform over time. In Figure 5.22, x-axis indicates time intervals, while y-axis indicates the number of distinct vehicles reaching the infrastructure. Opt_{dv} presents higher number of V2I contacts during the entire experiment.

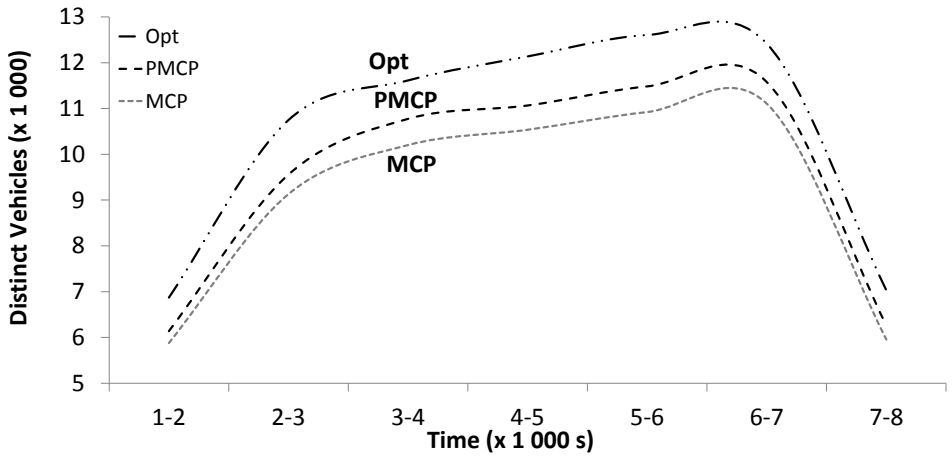


Figure 5.22. Contacts in Intervals of 1,000s. The x-axis indicates time intervals, while y-axis indicates the number of distinct vehicles reaching the infrastructure.

Table 5.5 presents average V2I contacts over time: PMCP-based achieves 90.5% of Opt_{dv}, while MCP-based achieves 85.1%.

Table 5.5. Distinct Vehicles: PMCP-based and Optimum Deployment

Distinct Vehicles per 1.000s	PMCP-based	MCP-based	Opt_{dv}
average	8,524	8,020	9,421
(% Opt_{dv})	90.5%	85.1%	100.0%
std. dev.	3,651	3,491	3,801

Figure 5.23 presents a similar study, but now we consider time intervals. The x-axis indicates the time interval, while the y-axis indicates how far each deployment

strategy is from Opt_{dv}. We measure the distance using distinct vehicles reaching the infrastructure. Shadowed area indicates standard deviation. PMCP-based clearly overcomes MCP-based during all the experiment.

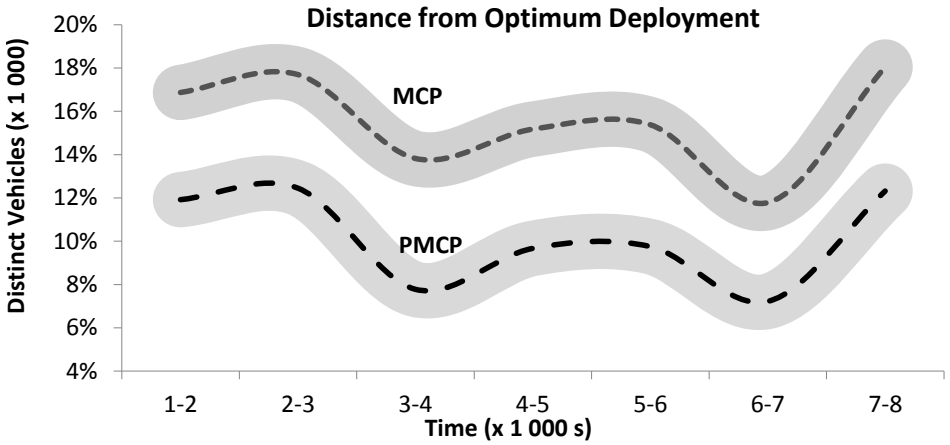


Figure 5.23. Distinct Vehicles: Distance from Opt_{dv}. The x-axis indicates time interval, while y-axis indicates the distance from PMCP-based and MCP-based to Opt_{dv} in terms of distinct vehicles reaching the infrastructure.

5.5.2 Distinct Vehicles Crossing Each Roadside Unit

Now we analyze the number of distinct vehicles crossing each roadside unit. In Figure 5.24 the x-axis indicates the roadside unit's ID, while the y-axis indicates the number of distinct vehicles crossing the roadside unit. Opt_{dv} shows an almost constant marginal contribution of each roadside unit, while PMCP-based and MCP-based demonstrates greediness. Opt_{dv} just overcomes PMCP-based and MCP-based after the 350^{th} roadside unit. Opt_{dv} demonstrates such behavior because our optimization model is not instrumented to 'save' roadside units: it just distributes the 500 roadside units in order to achieve maximum number of contacted vehicles.

5.5.3 Contact Time

Here we present the global vehicle-to-infrastructure contact time for PMCP-based, Opt_{dv}, and MCP-based. Global contact time refers to the sum of individual contact times for all vehicles. We consider this analysis relevant because contact time is not part of the objective function of Opt_{dv}. Obviously, Opt_{dv} performs better when comparing distinct vehicles. But what happens when we compare contact time? In Figure 5.25, x-axis indicates the RSU's ID, while y-axis indicates global vehicle-to-infrastructure

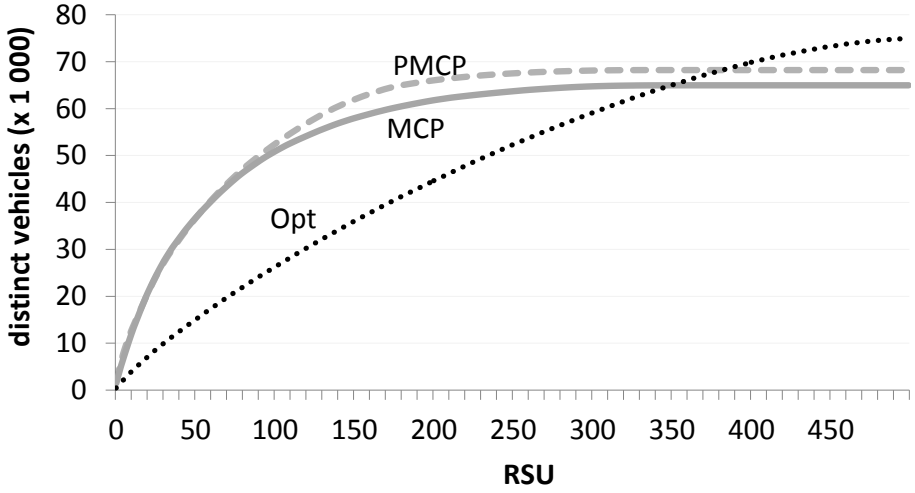


Figure 5.24. Distinct Vehicles per Roadside Unit (500 RSUs). The x-axis indicates the RSU's ID, while y-axis indicates the number of distinct vehicles crossing the roadside unit.

contact time. PMCP-based and MCP-based present higher contact time than Opt_{dv}, after all, Opt_{dv} is not optimized for maximizing global contact time. Because Opt_{dv} maximizes just the number of distinct vehicles reaching the infrastructure, Opt_{dv} has no commitment in optimizing contact time or another possible objective function. We notice that Opt_{dv} presents a high degradation of global V2I contact time when compared to PMCP-based and MCP-based.

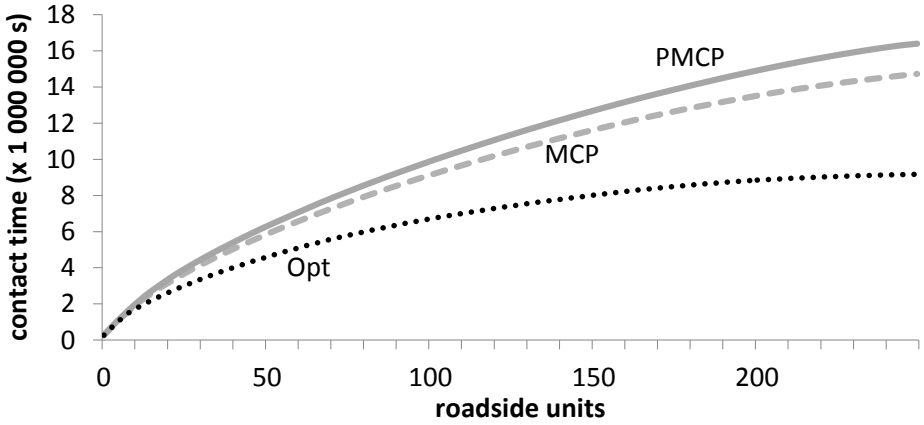


Figure 5.25. Vehicles Contact Time for 250 RSUs: x-axis indicates the RSU's ID, while y-axis indicates global vehicle-to-infrastructure contact time.

Figure 5.26 reveals the number of vehicles crossing each one of the deployed roadside units: x-axis indicates the roadside unit's ID, while y-axis indicates the number of vehicle-to-infrastructure contacts. Descending aspect of the plot demonstrates

greediness of PMCP-based. Figure 5.27 presents the same metric, now for Opt_{dv}. There is no apparent pattern describing the number of vehicles contacting the infrastructure. Our conclusion is marginal contribution of each roadside unit is better defined through PMCP-based: better results arise from the flow projection carried out in PMCP-based.

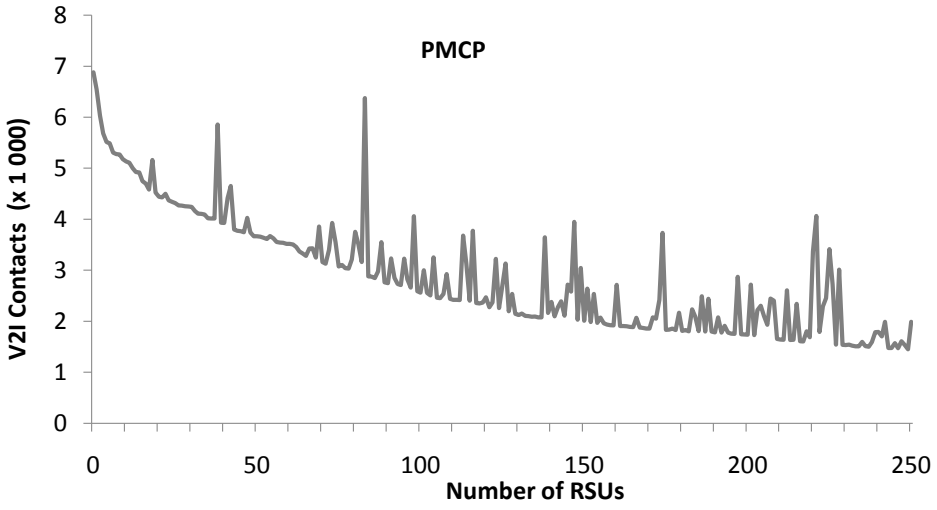


Figure 5.26. Contacts per Roadside Unit (250 RSUs): x-axis indicates the roadside unit's ID, while y-axis indicates the number of vehicle-to-infrastructure contacts.

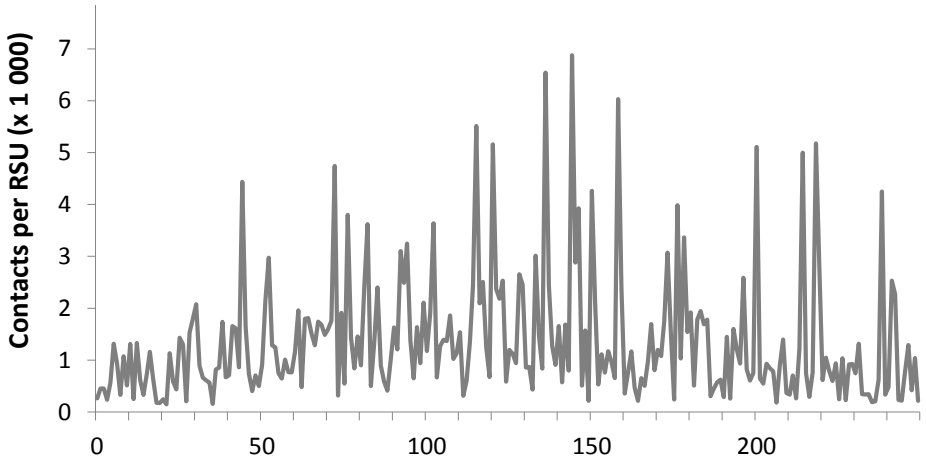


Figure 5.27. Contacts per Roadside Unit (250 RSUs): x-axis indicates the roadside unit's ID, while y-axis indicates the number of vehicle-to-infrastructure contacts.

5.5.4 Roadside Units Crossed per Vehicles

Here we analyze the number of roadside units crossed by vehicles considering PMCP-based and Opt_{dv}. In Figure 5.28, the x-axis indicates the number of roadside units crossed per vehicle (from 0 up to 10), while y-axis indicates the percentage of vehicles. Opt_{dv} seems to provide a better deployment when we consider the number of roadside units crossed per vehicle (from zero up to ten roadside units). Figure 5.29 presents the same analysis, but now considering the entire set of roadside units crossed. We notice that PMCP-based presents more vehicles experiencing several V2I contacts opportunities. Table 5.6 summarizes the experiment:

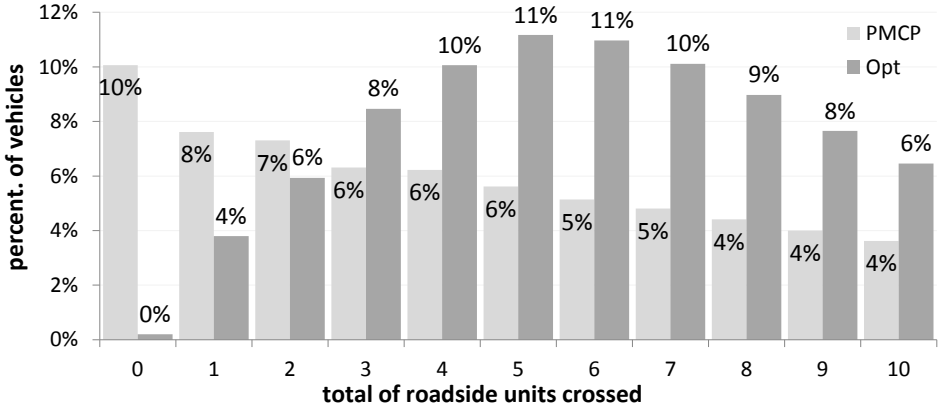


Figure 5.28. Roadside units crossed per vehicle: x-axis indicates the number of roadside units crossed per vehicle, while y-axis indicates the percentage of vehicles.

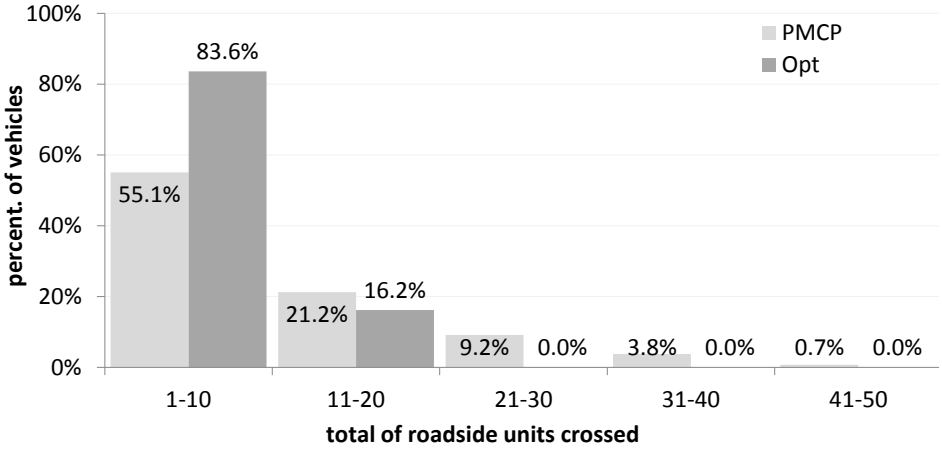


Figure 5.29. Roadside units crossed per vehicle: x-axis indicates the number of roadside units crossed per vehicle, while y-axis indicates the percentage of vehicles.

Table 5.6. Percentage of distinct vehicles reaching infrastructure. (*) Vehicles presenting very high number of contacts.

Number of Contacts	Opt_{dv}	PMCP-based
Uncovered Vehicles	0.2%	10.0%
1-10	83.6%	55.1%
11-20	16.2%	21.2%
21-30	0.0%	9.2% (*)
31-40	0.0%	3.8% (*)
41-50	0.0%	0.7% (*)

5.5.5 Distinct Grid Setups

Now we evaluate the number of distinct vehicles contacting the infrastructure for different grid setups. In Figure 5.30, x-axis indicates grid setup, while y-axis indicates the percentage of distinct vehicles contacting the infrastructure at least once. Plot represents the performance of PMCP-based and Opt_{dv}. Initially, we evaluate the effects of partitioning the road network by considering optimal deployment (Opt_{dv}): as we have already mentioned, partitioning the road network does not increase the deployment performance indefinitely. Partitioning the Cologne's road network in 40.000 urban cells (200x200-grid) ensures a performance gain of just 0.5% when we compare to partitioning the same road network in just 6.400 urban cells (80x80). Figure 5.30 also indicates the performance of PMCP-based: we can notice that PMCP-based presents a performance improvement when we partition the road network in a 200x200-grid, leading us to conclude that optimal number of partitions depends on the deployment algorithm adopted. We also notice that the performance of PMCP-based is very close to Opt_{dv} when we partition the road network in a 200x200 urban cells.

Table 5.7. Percentage of distinct vehicles reaching infrastructure.

Opt_{dv}		$PMCP_{dv}$	
Grid Setup	% Distinct Vehicles	Grid Setup	% Distinct Vehicles
20x20	88.6%	38x38	88.6%
30x30	93.5%	60x60	93.5%
40x40	96.5%	100x100	96.5%
50x50	97.8%	110x110	97.8%
60x60	98.9%	150x150	98.9%
100x100	99.8%	200x200	99.8%

In fact, an in-depth analysis of this experiment suggests that PMCP-based can be approximate to Opt_{dv} by increasing the number of partitions used by PMCP-based, as shown in Table 5.7. Although Table 5.7 may suggest that doubling grid setup is enough

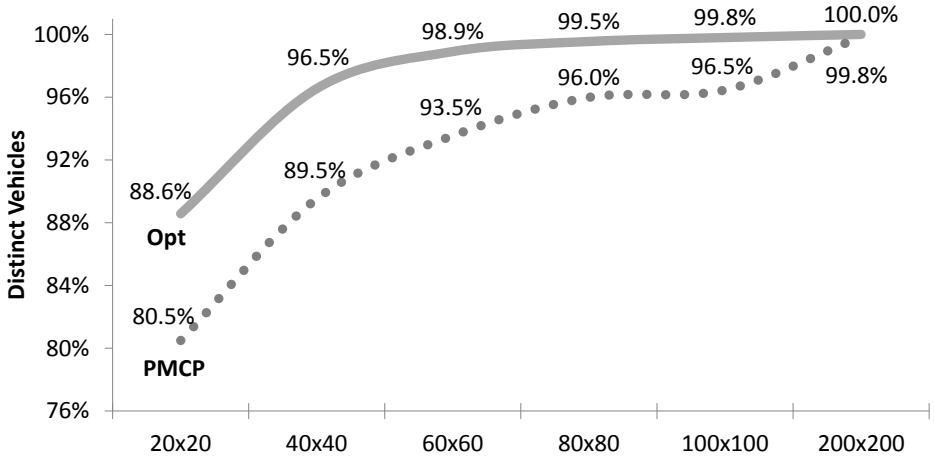


Figure 5.30. Distinct Vehicles Contacting Infrastructure: x-axis indicates grid setup, while y-axis indicates the percentage of distinct vehicles contacting the infrastructure at least once.

to approximate PMCP-based to Opt_{dv}, we highlight we do not intend to infer any relationship between grid setups of PMCP-based and Opt_{dv}: measurements reported above are meaningful only to this specific scenario. But the general idea certainly holds: increasing the number of urban cells improves the quality of PMCP-based, making it increasingly similar to Opt_{dv}. Also notice that increasing the number of partitions used by PMCP-based does not represent any issue, after all, processing time of PMCP-based is very small when compared to Opt_{dv}. Next section analyzes the processing time of the deployment algorithms.

5.5.6 Processing Time

Here we present computation time required to process the Cologne scenario. Figures 5.31 and 5.32 present processing time for PMCP-based and MCP. The x-axis indicates grid setup, while y-axis indicates time in seconds. IBM CPLEX was configured to a time limit of 21 600s (6h).

5.6 Full Projection of the Flow

In this chapter we extend our projection of the flow to consider the entire urban area. Our strategy is based on the **Full Projection of the Flow** (FPF). Every time a new roadside unit is added we recompute the number of uncovered vehicles along the entire road network using a trajectories prediction scheme supported by the migration ratios between adjacent urban cells. We evaluate our deployment algorithm considering

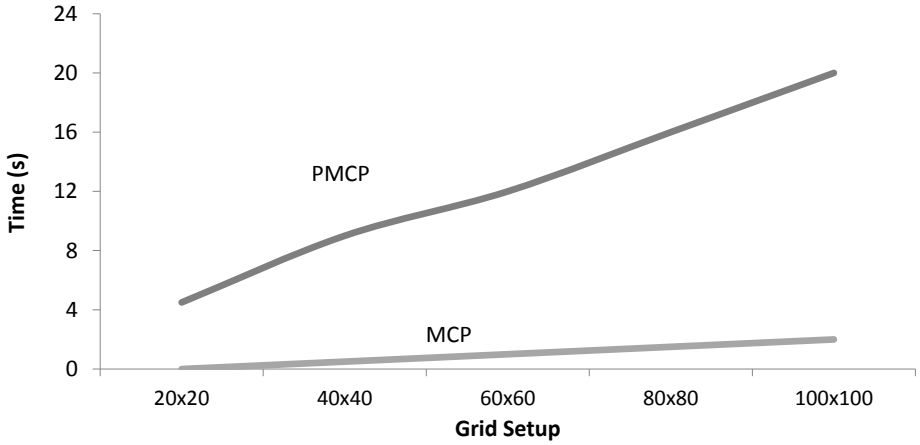


Figure 5.31. Processing time for PMCP-based and MCP: x-axis indicates grid setup, while y-axis indicates time in seconds.

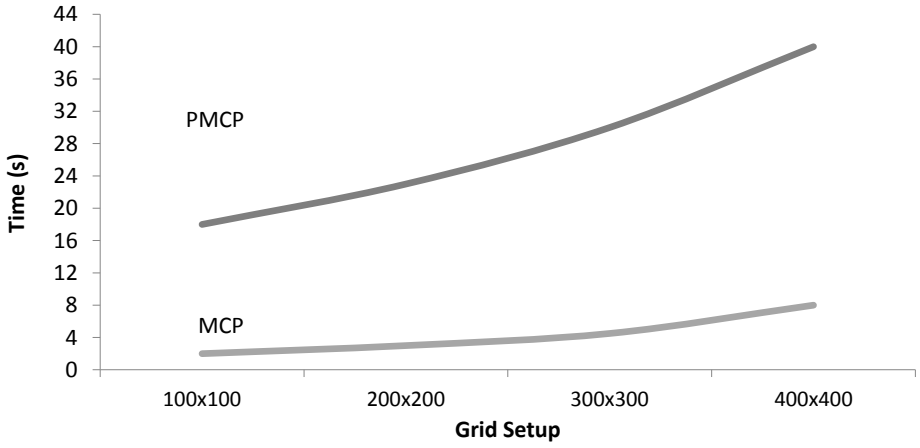


Figure 5.32. Processing time for PMCP-based and MCP: x-axis indicates grid setup, while y-axis indicates time in seconds.

realistic conditions (real road network and realistic flow), and we compare FPF to the greedy solution of the Maximum Coverage Problem, a close-to-optimal deployment strategy relying on full knowledge of vehicles trajectories (optimal solution is at most 3.5% higher than the greedy solution). Our results demonstrate that:

- Previously knowledge of the vehicles trajectories is not mandatory for achieving a close-to-optimal deployment performance: by covering 1.0% of the Cologne's road network, our FPF deployment yields 89.8% of all vehicles experiencing at least one V2I contact opportunity. On the other hand, if we assume previous knowledge of individual vehicles trajectories, we improve the performance in just 2.3%.

- If we double the covered area (from 1% to 2%), we improve the vehicles coverage in less than 6%. Thus, covering just 1% of the entire road network is sufficient for distributing small traffic announcements;

- Our strategy provides a deployment layout very similar to the one obtained when considering the individual vehicles trajectories.

This section is organized as follows: Section 5.6.1 presents FPF, our deployment strategy based on the full projection of the flow. Section 5.6.2 presents our experiments considering a real road network and the realistic vehicular trace of the Cologne's city. Section **??** concludes the chapter.

5.6.1 FPF: Full Projection of the Flow Deployment

The FPF strategy employs a trajectories prediction scheme in order to select the urban cells that must receive the roadside units with the goal to maximize the number of distinct vehicles experiencing at least one V2I contact opportunity. FPF locates the roadside units at urban cells presenting the highest number of (yet) uncovered[7] vehicles. The goal is to identify the number of uncovered vehicles along the entire road network. Initially, we select the urban cell presenting the highest number of vehicles. Such urban cell receives a roadside unit. After adding the roadside unit, we compute the number of vehicles crossing this newly deployed roadside unit. These vehicles are removed from our model (since they will cross a roadside unit soon). Then, again we select the location presenting the highest number of uncovered vehicles, and we repeat interactively until we have selected the total of α urban cells.

Analytically, we may represent the projection of the flow using Equations 5.7 and 5.8: Equation 5.7 excludes from $M_{i,j}$ vehicles crossing {i,j} and then {x,y}, where x and y are the grid coordinates of the urban cell receiving the new roadside unit. Equation 5.8 excludes from $M_{i,j}$ vehicles crossing {x,y} and then {i,j}.

$$M_{i,j} \leftarrow M_{i,j} \times (1 - P_{i,j,x,y}) \tag{5.7}$$

$$M_{i,j} \leftarrow M_{i,j} - M_{x,y} \times P_{x,y,i,j} \tag{5.8}$$

[7]We are assuming a slightly different concept of 'coverage': traditional usage of coverage indicates a continuous region where users are supposed to meet connection. But, because we are assuming **infostations**, we consider small islands of coverage: fragmented and possibly disconnect areas where users are supposed to meet connection.

FPF strategy is presented in Algorithm 9. It receives as input the bi-dimensional matrix M of urban cells representing the flow within a partitioned road network, the number of available roadside units (α), the four dimensional matrix of migration ratios P, and the grid setup (ϕ). Each $M_{i,j}$ indicates the number of vehicles crossing the urban cell $\{i, j\}$ during the observation period. Each $P_{i,j,k,l}$ indicates the migration ratios between the urban cells $\{i, j\}$ and $\{k, l\}$.

Algorithm 9 FPF: Full Projection of the Flow Deployment.

Input: M, α, P, ϕ;
Output: Γ (solution set);

```
1: Γ ← ∅;
2: m ← Get_Cell_Presenting_Max_Vehicles(M);
3: Γ ← Γ ∪ m;
4: for i=2 to α do
5:     Full_Flow_Projection(m.x,m.y);
6:     m ← Get_Cell_Presenting_Max_Projection(M);
7:     Γ ← Γ ∪ m;
8: end for
9: return Γ;
10: procedure Full_Flow_Projection(x,y)
11:     for i=1 to φ do
12:         for j=1 to φ do
13:             M_{i,j} ← max(M_{i,j} × (1 − P_{i,j,x,y}), 0);
14:             M_{i,j} ← max(M_{i,j} − M_{x,y} × P_{x,y,i,j}), 0);
15:         end for
16:     end for
17: end procedure
```

FPF iteratively selects the urban cells presenting the highest projections of the flow. At the first iteration, FPF selects the densest urban cell. Such urban cell is added to the solution set Γ. When it selects an urban cell, FPF recomputes the number of distinct-and-uncovered vehicles for all remaining urban cells using the migration ratios P. Function Full_Flow_Projection receives the coordinates of the selected urban cell ({x,y}), and it removes vehicles that will cross the new roadside unit. Line 13 remove vehicles that cross {i,j} and then {x,y}. Line 14 remove the vehicles that cross {x,y}, and then {i,j}. The computational complexity of FPF is $\Theta(\alpha.M)$.

5.6.2 Evaluation On a Realistic Scenario

Now we present a set of experiments comparing the performance of FPF, MCP-g, and MCP-kp considering the realistic mobility trace of Cologne's city (Germany) composed

of approximately 10 000s of traffic and 75 515 vehicles. Our goal is to quantify the impact of the mobility information (none, partial, and full) on their performance. Recall that MCP-kp lacks any mobility information: it just sorts the urban cells and selects those α urban cells presenting the highest volume of vehicles. On the other hand, FPF considers partial mobility information (migration ratios between adjacent urban cells): it starts by picking the most crowded urban cell. Then, it projects the flow of vehicles according to the stochastic matrix of migration ratios (P), and it selects the urban cell presenting the highest expectancy of vehicles. Then, it projects the flow once again, and selects another urban cell. FPF repeats this process α times. Finally, MCP-g considers full mobility information (trajectories of vehicles): it always selects the urban cells containing the highest number of uncovered vehicles.

5.6.2.1 Methodology

All experiments are performed using the SUMO[8] simulator and a set of tools designed by our team. SUMO runs the Cologne scenario and outputs the location of each vehicle (our mobility trace T) over time. The Partition Program reads the mobility trace, computes the bounding box of the mobility trace, partitions the Cologne into a grid of $\psi \times \psi$ urban cells, and then translates the mobility trace from Cartesian coordinates to Grid coordinates.

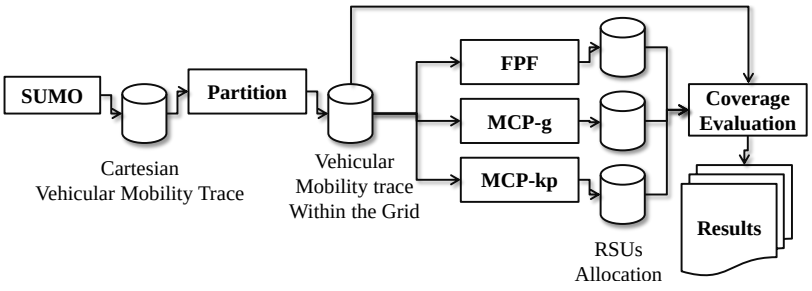


Figure 5.33. Simulation Work Flow.

After that, we run Programs MCP-g, MCP-kp, and FPF: each of these programs outputs an individual file containing the position of the roadside units according to each strategy. Then, we run the Coverage Evaluation Program: it receives as input the mobility trace and a file containing the location of the α roadside units, and outputs several reports used to evaluate the deployment performance. We have validated the Program Flow by generating an Integer Linear Programming Formulation (Dantzig,

[8]Sumo Simulator: `http://sumo-sim.org`.

1998) of MCP-g for the IBM CPLEX Optimizer[9]. IBM CPLEX outputs the location of the roadside units and the status of all vehicles (covered or non-covered). When we run the Program MeasureFlow using the IBM CPLEX output we achieve the exact same results. Programs MCP-g, MCP-kp, and FPF are validated through the inspection of small scenarios. Figure 5.33 illustrates our work flow simulation.

The experiments presented in this work are based on the Cologne's city (Germany). We partition the road network into 100×100 grid (ψ=100) in order to achieve urban cells able to be potentially covered by just one roadside unit. A 100×100 grid partitions the Cologne's road network into urban cells of approximately 270m x 260m, a typical range assumed to present high contact probability: the work (Teixeira et al., 2014) presents a field test evaluation of the IEEE 802.11p standard. Authors report an average bit rate of 3Mbps when considering 500 bytes UDP packets, and 8Mbps for 1 460 bytes packets. Packet loss is around 1% for distances below 300m. Thus, the selection of a 100×100 grid is realistic.

5.6.2.2 Covered Vehicles x Covered Area

Figure 5.34 presents the percentage of vehicles experiencing at least one V2I contact opportunity when we deploy the roadside units according to FPF, MCP-kp, or MCP-g. The x-axis shows the percentage of covered area (and the number of deployed roadside units), while the y-axis indicates the percentage of covered vehicles.

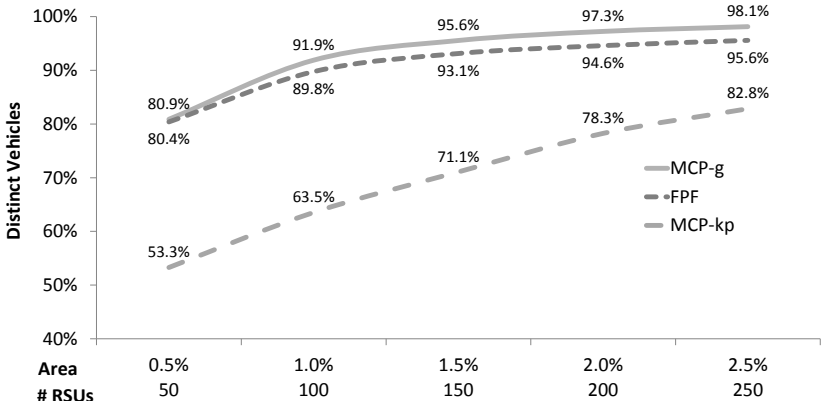


Figure 5.34. Percentage of Distinct Covered Vehicles × Covered Area. The x-axis shows the number of deployed roadside units. The y-axis indicates the percentage of vehicles experiencing at least one V2I contact opportunity.

By covering 1.0% of Cologne (selection of 100 urban cells in a set of 10 000), we reach the following percentage of vehicles: MCP-g=91.9%; FPF=89.8%;

[9]IBM CPLEX Optimizer: `http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/`

MCP-kp=63.5%. FPF improves MCP-kp in 41.3%. Recall that the only distinction between FPF and MCP-kp is the migration ratios information. Complementary, MCP-g requires the full vehicles trajectories in order to improve FPF in just 2.3%. **Such result clearly demonstrates that the trajectories information are not necessary for achieving a good deployment performance**. Furthermore, when we increase the covered area in 100%, we improve MCP-g in +5.8%, FPF in +5.4%, and MCP-kp in +23.1% (Table 5.8).

Table 5.8. Covered Area x Covered Vehicles.

	Scen. #1	Scen. #2	Improvement
Covered Area	1.0%	2.0%	100%
Deployed Roadside Units	100	200	100%
MCP-g	91.9%	97.3%	+5.8%
FPF	89.8%	94.6%	+5.4%
MCP-kp	63.5%	78.3%	+23.1%

Thus, in the following experiments we consider the deployment of 100 roadside units covering 1.0% of the Cologne's road network.

5.6.2.3 Coverage Improvement Analysis

Now we highlight the percentage improvements of both strategies: Figure 5.35 plots the improvements of FPF over MCP-kp (Relation i_1), and the improvements of MCP-g over FPF (Relation i_2). Relation i_1 is given by $\frac{FPF}{MCP_{kp}} - 1$, and it peaks at 50.9% when x-axis is 0.5%. Relation i_2 is $\frac{MCP_g}{FPF} - 1$, and it does not present any peak. When we increase the number of roadside units, MCP-kp gradually approximates to FPF and MCP-g: beyond 100 roadside units, both FPF and MCP-g become saturated: remaining uncovered vehicles travel at locations where the flow is minimal.

5.6.2.4 Low Scale Deployment

In this experiment we analyze the infrastructure performance for a low-scale-incremental deployment. Figure 5.36 presents an incremental view of the deployment considering small increments of 10 newly deployed roadside units.

FPF and MCP-g show almost the same performance in terms of distinct vehicles reaching the infrastructure, and MCP-kp shows a poor performance. Such issue demonstrates quantitatively that placing the roadside units at the densest locations of the road network (without taking the mobility into account) is not the best option.

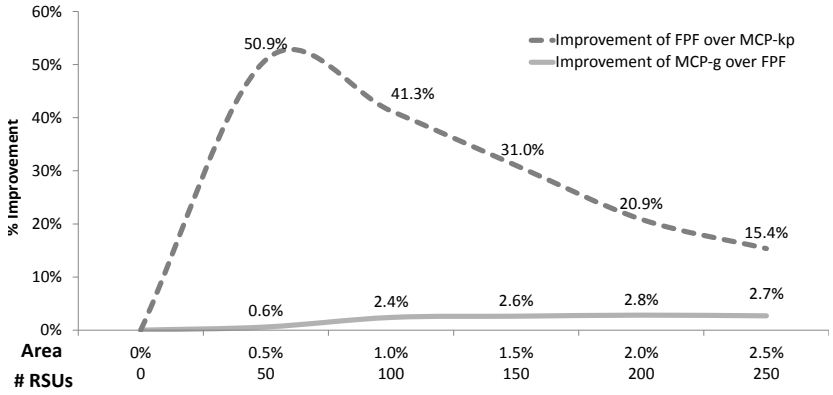


Figure 5.35. Improvements of MCP-g over FPF, and FPF over MCP-kp in terms of the trips experiencing at least one V2I contact opportunity. The x-axis indicates the number of deployed roadside units. The y-axis indicates the percentage improvement.

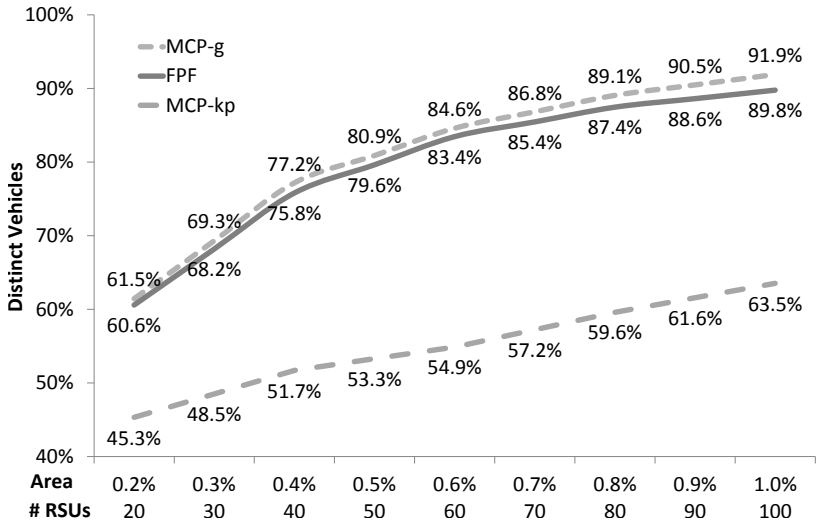


Figure 5.36. Low Scale Deployment. The x-axis indicates the number of deployed roadside units. The y-axis indicates the percentage of vehicles experiencing at least one V2I contact opportunity.

5.6.2.5 Number of V2I Contacts per Roadside Unit

In Sections 5.6.2.2 to 5.6.2.4 we have characterized the number of distinct vehicles experiencing at least one V2I contact opportunity. Now we characterize the total number of V2I contact opportunities (one vehicle may contact the infrastructure more than once during the trip).

Figure 5.37 plots the number of V2I contact opportunities per roadside unit considering the MCP-kp deployment. The x-axis indicates the ID of the roadside unit (sequentially from 0 up to 99 following the deployment ordering), while the y-axis

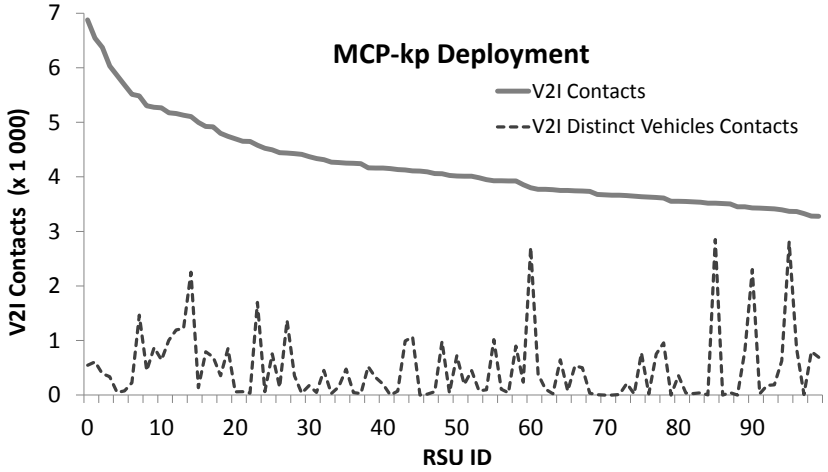


Figure 5.37. MCP-kp: Number of V2I contacts per RSU. The x-axis presents the RSUs in the order of deployment. The y-axis indicates the total number of V2I contacts for each RSU. Plot shows the total V2I contacts (continuous line) and the number of V2I distinct vehicles contacts.

indicates the number of V2I contact opportunities. Continuous line indicates the total number of V2I contacts for each roadside unit. Since MCP-kp selects the densest urban cells for deploying the roadside units, the number of V2I contacts decreases as we increase the roadside units' ID (the first roadside unit is placed at the densest urban cell, the second roadside units is placed on the second densest urban cell, and so forth). Dotted line indicates the number of distinct vehicles contacting the infrastructure. We consider just the first V2I contact opportunity of each vehicle. We notice some roadside units reaching a high number of vehicles, while others do not reach any distinct vehicles at all (e.g., RSU #70). Each roadside unit covers a median of 220 ± 122 vehicles considering a confidence interval of 95%.

Figure 5.38 presents the same analysis, but now considering a MCP-g deployment. The continuous line indicates that the number of total V2I contacts presents high variance for distinct roadside units. On the other hand, the number of distinct vehicles contacting the roadside units (dotted line) presents less variance than MCP-kp, after all, MCP-g selects the urban cells according to the number of uncovered vehicles. Each roadside unit covers a median of 619 ± 57 vehicles considering a confidence interval of 95%.

Figure 5.39 presents the same analysis, but now considering a FPF deployment. The dotted line indicates a greedy behavior for FPF. Each roadside unit covers a median of 532 ± 84 vehicles considering a confidence interval of 95%.

Table 5.9 summarizes such results.

This experiment demonstrates that relying on individual vehicles trajectories

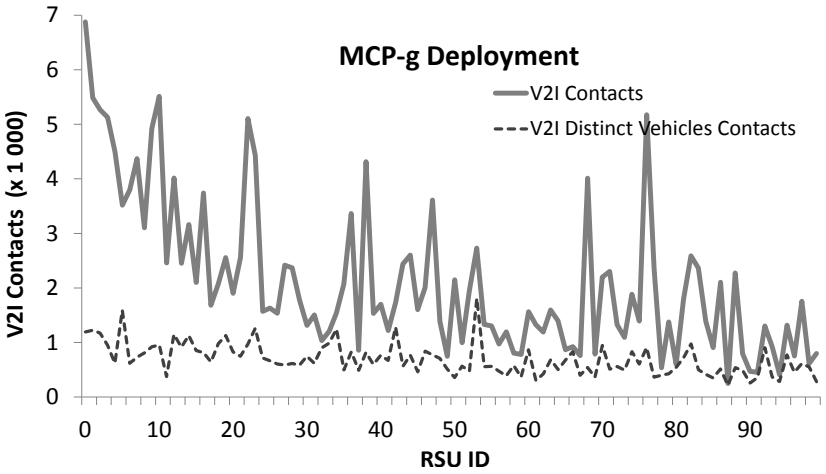


Figure 5.38. MCP-g: Number of V2I contacts per RSU. The x-axis presents the RSUs in the order of deployment. The y-axis indicates the total number of V2I contacts for each RSU. Plot shows the total V2I contacts (continuous line) and the number of V2I distinct vehicles contacts.

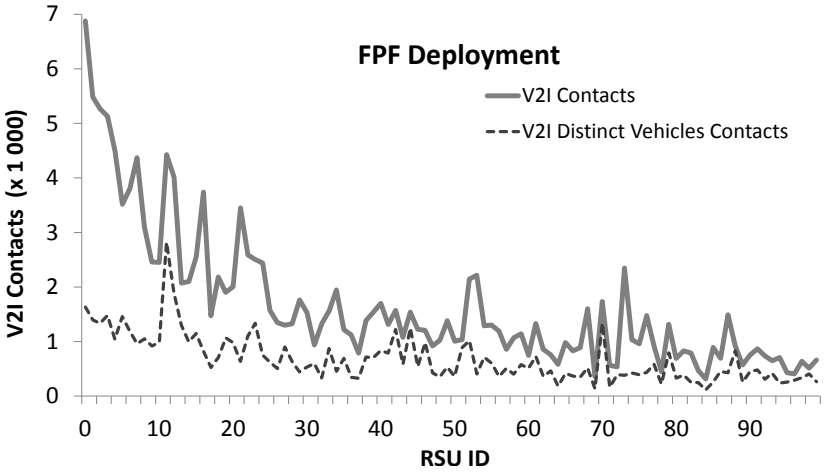


Figure 5.39. FPF: Number of V2I contacts per RSU. The x-axis presents the RSUs in the order of deployment. The y-axis indicates the total number of V2I contacts for each RSU. Plot shows the total V2I contacts (continuous line) and the number of V2I distinct vehicles contacts.

provides a small increase in the deployment performance (from 677 ± 84 to 694 ± 57 distinct vehicles per roadside unit). When we consider total number of contacts, MCP-kp achieves 796 021 contacts, followed by FPF with 222 049 (-72.1%) and MCP-g with 371 647 (-53.3%). FPF reduces the contact opportunities in 72.1% when compared to MCP-kp. On the other hand, it increases the number of distinct vehicles being covered (from 63.5% to 89.8%).

Table 5.9. Distinct-and-Covered Vehicles per Roadside Unit.

	Average	Median	Confidence (95%)
MCP-kp	486	220	122
FPF	677	532	84
MCP-g	694	619	57

5.6.2.6 Number of Roadside Units Crossed per Vehicle

As we have previously mentioned, the goal of our strategy is to reduce the redundant coverage in order to better distribute the contact opportunities. Therefore, in Figure 5.40 we present the number of roadside units crossed per vehicle considering the three deployment strategies. The x-axis indicates the number of crossed roadside units. The y-axis indicates the number of vehicles crossing that amount of roadside units (notice we use a logarithmic scale in the y-axis).

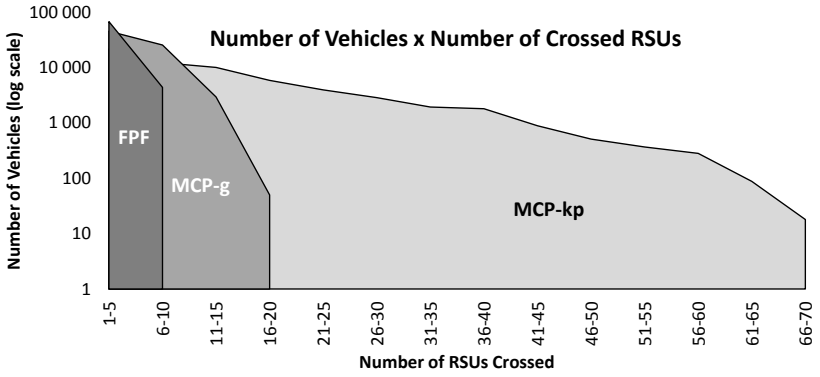


Figure 5.40. Number of crossed RSUs per vehicle. The x-axis indicates the amount of RSUs crossed. The y-axis indicates the number of vehicles ($\log_{10}$).

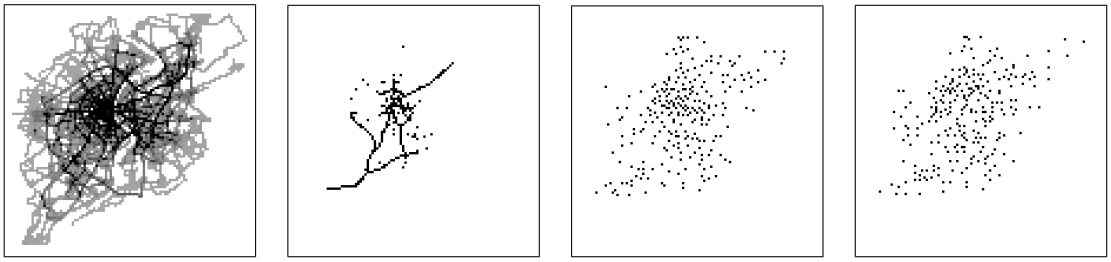
- FPF vehicles reach 10 roadside units at most.
- MCP-g vehicles reach 20 roadside units at most.
- MCP-kp vehicles reach up to 70 roadside units.

MCP-kp presents more vehicles driving through more roadside units because it concentrates roadside units in very popular locations, and vehicles crossing such popular locations experience several contact opportunities. The strategy to concentrate roadside units in very popular locations also has a side-effect: 36.5% of the MCP-kp vehicles never cross any roadside unit. For a matter of comparison, the same measure

for FPF is 10.2%, while for MCP-g it is just 8.1%. Thus, FPF is able to reduce the redundant coverage provided by MCP-kp, but MCP-g provides a better distribution of V2I contacts.

5.6.3 Roadside Units Layout

Now we present the layout of the roadside units deployed according to each strategy. The 100×100 partitioned map of the Cologne's city is presented in Figure 5.41(a) (the darkest the area, the more intense is the flow). MCP-kp deployment is indicated in Figure 5.41(b) (black dots represents roadside units). As we can notice, the roadside units are allocated at very popular urban cells. Vehicles traveling these very popular routes experience several contact opportunities.



(a) Road network. (b) MCP-kp. (c) MCP-g. (d) FPF.

Figure 5.41. Road network and layout of roadside units for: MCP-kp, MCP-g, and FPF.

MCP-g deployment is presented in Figure 5.41(c), a very distinct layout from MCP-kp. MCP-g deploys the roadside units in order to reach the maximum number of distinct vehicles, so the roadside units are spread along the entire road network. Finally, the FPF deployment is presented in Figure 5.41(d), and we notice a high similarity to MCP-g.

5.7 Remarks

In this chapter we propose and evaluate a deployment algorithm based on migration ratios between adjacent urban cells without relying on individual vehicles trajectories. We call it a deployment based on partial mobility information. We compare our approach to MCP-kp and MCP-g, both heuristics proposed by Trullols et al. Trullols et al. (2010). MCP-g is the greedy heuristic for Maximum Coverage Problem and it requires full knowledge of vehicles trajectories. We call it a full mobility information.

On the other hand, MCP-kp deploys the roadside units at locations presenting the highest concentration of vehicles. Our goal is to maximize the number of distinct vehicles experiencing at least one V2I contact opportunity.

MCP-g provides a better deployment since it counts on with vehicles trajectories. However, such information may not be available on a real deployment. Thus, we have turned our attention to evaluate the use of partial mobility information based on the global behavior of drivers. We perform an in-depth comparison among the three strategies considering the realistic vehicular mobility trace of the Cologne's city. When we analyze the step-by-step deployment presented in Section 5.2.1, we notice that PMCP-b may be improved by expanding the projection of the flow. In this work we have defined the projection of the flow to recompute the number of uncovered vehicles located at the same row or at the same column of the selected urban cell.

We also propose an Integer Linear Programming Formulation (Opt_{dv}) for solving the deployment of roadside units. Model runs on IBM CPLEX. Our main findings are:

- PMCP-based achieves an average of 90.5% of Opt_{dv}, while MCP achieves 85.1% (Section 5.5.1);
- Opt_{dv} shows an almost constant marginal contribution of each roadside unit in terms of distinct vehicles, while PMCP-based and MCP-based demonstrates greediness (Section 5.5.2);
- PMCP-based presents 60% more contact time than Opt_{dv}. Because Opt_{dv} maximizes just the number of distinct vehicles reaching the infrastructure, the optimization model has no commitment in optimizing contact time or another possible objective function (Section 5.5.3);
- Marginal contribution of global contact time when adding new roadside units is more precise in PMCP-based (Section 5.5.3);
- When we consider the number of roadside units crossed per vehicle, Opt_{dv} demonstrates more regularity. However, besides the regularity provided by Opt_{dv}, PMCP-based presents vehicles contacting much more the infrastructure, which is desirable for the deployment of infrastructure (Section 5.5.4);
- Partition does not increase deployment performance indefinitely: partitioning the Cologne's road network in 40.000 urban cells (200x200-grid) ensures a performance gain of just 0.5% when we compare to partitioning the same road network in just 6.400 urban cells (Section 5.5.5);

- Optimum number of partitions depends on flow characteristics, deployment algorithm and the road network itself (Section 5.5.5);
- PMCP-based can be approximate to Opt_{dv} by increasing the number of partitions used by PMCP-based. Increasing the number of partitions for PMCP-based does not represent an issue: processing time of PMCP-based is very small when compared to Opt_{dv} (Section 5.5.5);
- Time required to process the Cologne scenario in MCP-based is less than 10s using an ordinary 4GB RAM laptop. PMCP-based requires less than 40s. IBM CPLEX was executed with a time limit of 2hs (Section 5.5.6).

Finally, we propose FPF to extend the projection of the flow to the entire road, and our results demonstrate that:

- Previously knowledge of the vehicles trajectories is not mandatory for achieving a close-to-optimal deployment performance: by covering 1.0% of the Cologne's road network, our FPF deployment yields 89.8% of all vehicles experiencing at least one V2I contact opportunity. On the other hand, if we assume previous knowledge of individual vehicles trajectories, we improve the performance in just 2.3%.
- If we double the covered area (from 1% to 2%), we improve the vehicles coverage in less than 6%. Thus, covering just 1% of the entire road network is sufficient for distributing small traffic announcements;
- Our strategy provides a deployment layout very similar to the one obtained when considering the individual vehicles trajectories;

5.8 Overview of the Next Chapter

In the next chapter we evaluate a hybrid deployment consisting of stationary and mobile roadside units.

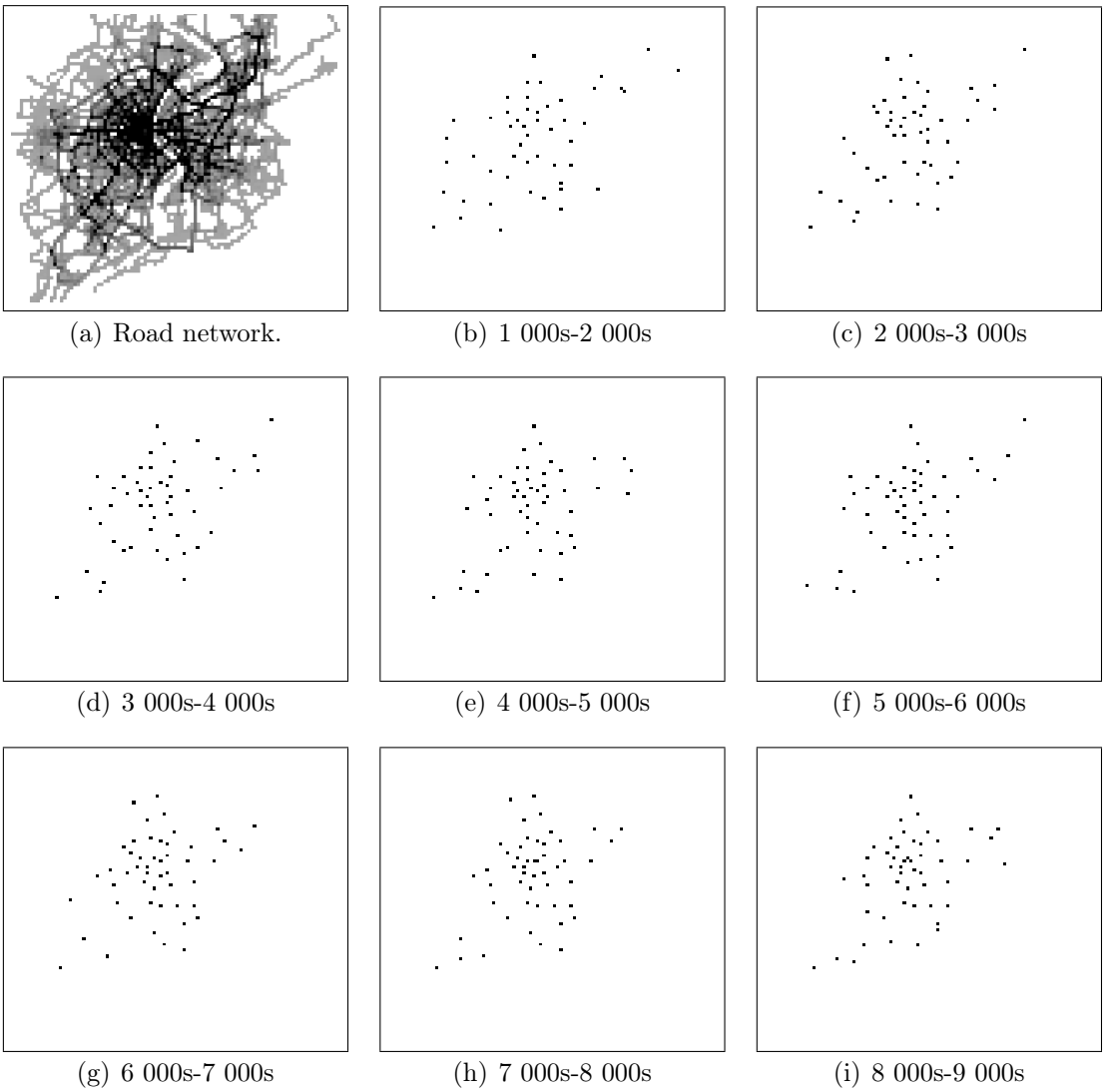
Chapter 6

Hybrid Deployment

During the last years several works have devoted their attention in proposing new strategies for infrastructure deployment over vehicular networks. Most of the works consider the deployment of stationary roadside units, while others (Tonguz and Viriyasitavat, 2013; Jerbi et al., 2008; Luo et al., 2010; Annese et al., 2011; Mishra et al., 2011; Sommer et al., 2013) propose the deployment of mobile ones. It is a common sense that traffic fluctuates: thus, an architecture employing just stationary roadside units might not be able to properly support the network operation all the time. Similarly, an architecture composed just of mobile roadside units makes a few sense when we consider that the road network does not change that often. In other words, traffic fluctuations are limited by its subjacent road network. As major roads counts on a higher transportation capacity, they tend to be very popular routes, and they are natural candidates for receiving stationary roadside units. And mobile roadside are highlighted as ideal solutions to handle in-borders traffic. In this chapter we investigate the benefits of adopting a hybrid architecture of roadside units composed of stationary and mobile roadside units.

In order to improve the deployment performance, we may rely on hybrid deployment strategies employing sets of stationary and mobile roadside units. Stationary roadside units act as a main backbone for data dissemination covering the most important regions of an urban area (i.e., regions known as always presenting relevant traffic). On the other hand, we can rely on mobile roadside units to address traffic variations. According to Weijermars (2007), temporal variations in traffic volumes can be analyzed at different time scales, ranging from minute-to-minute variations to year-to-year variations, and traffic presents variations according to: (i) type and time of day; (ii) weather conditions; (iii) events; (iv) road works; (v) accidents.

Figure 6.1 illustrates the use of a hybrid architecture covering the Cologne's city. Figure 6.1(a) presents the Cologne's map. Figures 6.1(b)-6.1(i) presents the layout of roadside units covering 0.5% of the road network. Each layout is computed using a time window of 1 000s of the mobility trace. An in-depth inspection reveals that some roadside units remain in the same position, while others present a slightly repositioning.



(a) Road network. (b) 1 000s-2 000s (c) 2 000s-3 000s

(d) 3 000s-4 000s (e) 4 000s-5 000s (f) 5 000s-6 000s

(g) 6 000s-7 000s (h) 7 000s-8 000s (i) 8 000s-9 000s

Figure 6.1. Layout of Roadside Units: Figure shows the layout of roadside units for deployments considering time windows of 1 000s.

Our results demonstrate that:

- Number of distinct vehicles contacting the infrastructure is maximized in static deployment (93.5%), while hybrid deployment is 83.2%;

- Number of vehicles contacting the infrastructure is maximized in hybrid deployment (69%), while static deployment is 59%;
- Hybrid deployment may improve static deployment up to 45%, depending on the time window under analysis;
- During adjacent time windows, 58% of the roadside units deployed by hybrid deployment preserves their positions;
- Hybrid deployment increases V2I contact time up to 18% when compared to static deployment;
- In order to implement hybrid deployment, the mobile roadside units must travel at speeds ranging from 5.2km/h up to 11.3km/h.

6.1 Deployment Algorithm

We investigate the use of hybrid architectures by modeling the deployment of roadside units as a Maximum Coverage Problem (Cormen et al., 2001) using the well-know greedy solution for MCP presented in Algorithm 1. Greedy heuristic achieves an approximation factor $(1 - 1/m)^m$, where m is the maximum cardinality of the sets in the optimization domain. Recall that heuristic MCP-g chooses sets (i.e., locations) according to one rule: at each stage, choose a set which contains the largest number of uncovered elements. MCP-g requires previous knowledge of vehicles trajectories. We have selected MCP-g for this investigation because it relies on full mobility information. Thus, MCP-g is an adequate strategy to exploit the dynamics of a hybrid architecture.

We evaluate three distinct deployment strategies:

- **Stationary Deployment:** Algorithm evaluates the entire mobility trace, and then it locates the stationary roadside units;
- **Hybrid Windowed Deployment (HWD):** Algorithm evaluates the position of each roadside unit periodically. We have arbitrarily defined a time window of 1 000s;
- **Hybrid Cumulative Deployment (HCD):** Algorithm also evaluates the position of each roadside unit periodically, but the time windows extends from the beginning of the mobility trace up to the actual moment. Thus, we gradually review the 'past' in order to reconfigure the position of the roadside units aiming to maximize covered trips. We have arbitrarily defined time steps of 1 000s.

6.2 V2I Contacts for Distinct Grid Setups

In Figure 6.2 we present the number of vehicles presenting at least one V2I contact opportunity when we partition the road network into several grid setups. The x-axis indicates the grid setup, while the y-axis indicates the percentage of vehicle experiencing one (or more) contact opportunities. We have deployed roadside units covering 0.5% of the entire road network employing MCP heuristic. We notice that over-partitioning does not increase the deployment efficiency indefinitely. Based on this result, we propose the partition of the Cologne's road network into a 100x100 grid.

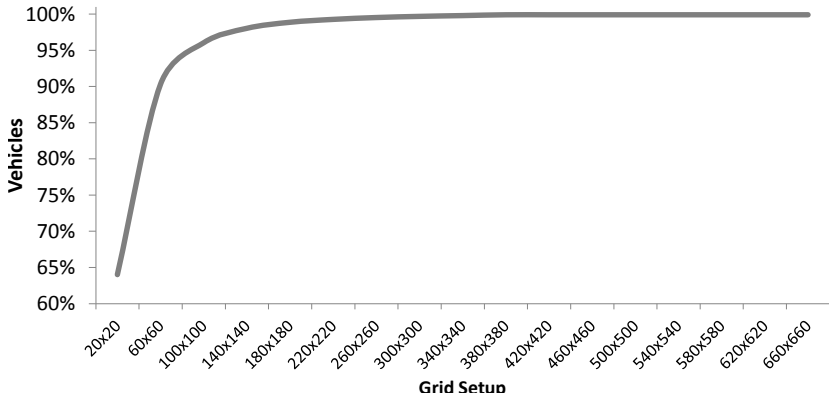


Figure 6.2. Vehicles Presenting V2I Contact Over Time.

Our results demonstrate that a 100x100 grid represents a good compromise between processing efforts and deployment accuracy. Thus, for the remaining experiments we partition the road network into 10 000 urban cells, and we select those 50 most promising cells. By covering just 0.5% of the road network, MCP-g achieves up to 93.5% vehicles experiencing at least one V2I contact opportunity. Figure 6.2 demonstrates that a proper grid resolution does not degenerate the deployment performance.

6.3 Characterizing the Cologne Scenario

Initially, we characterize the Cologne scenario. Figure 6.3 presents the number of new vehicles joining the simulation every 1 000s.

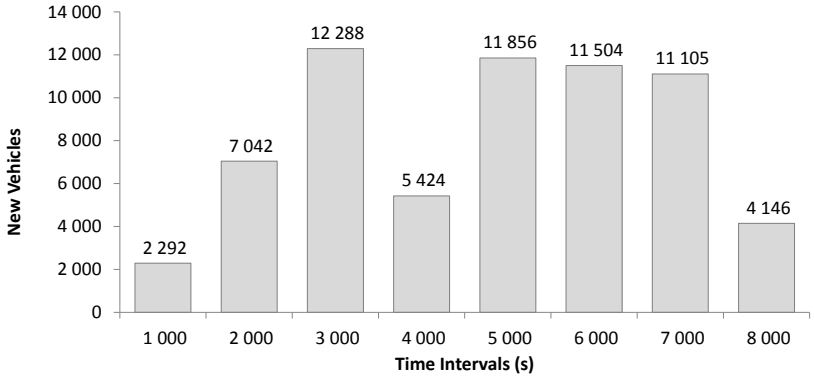


Figure 6.3. Vehicles Arrival: Figure presents the number of new vehicles joining the Cologne scenario each 1 000s.

6.4 Global V2I Contact Opportunities

Here we present a comparison between the three deployment strategies. Figure 6.4 presents the percentage of distinct vehicles experiencing at least one V2I contact opportunity during the simulation. We notice that Static Deployment overcomes Hybrid Windowed Deployment, and the reason is simple: our Hybrid Windowed Deployment places the roadside units maximizing the number of distinct vehicles covered during the time window, i.e., Hybrid Windowed Deployment does not keep track of vehicles covered during earlier time windows. We consider every time window as a new deployment. On the other hand, Static Deployment maximizes the number of uncovered vehicles by inspecting the entire mobility trace in a single step. Thus, Static Deployment will always offer a number of distinct vehicles greater than or equal to Hybrid Windowed Deployment.

Figure 6.4 presents our results for Static Deployment, Hybrid Cumulative Deployment, and Hybrid Windowed Deployment. The x-axis indicates time (s), while the y-axis indicates the number of vehicles experiencing at least one V2I contact opportunity. Percentages are computed considered the total number of vehicles participating the simulation (65 657), although not all of them are active all the time.

Note that Static Deployment and Hybrid Cumulative Deployment present almost the same values, thus both plots are superimposed. Next experiment evaluates the number of vehicles experiencing at least one contact opportunity within a given time window.

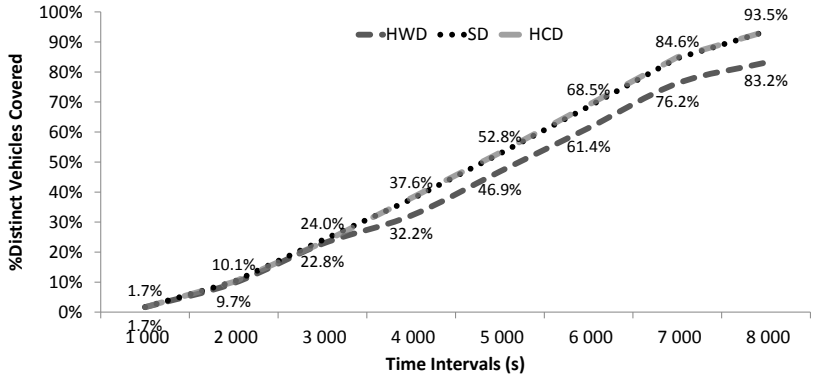


Figure 6.4. Global V2I Contact Opportunities: Figure presents the number of 'active vehicles' × 'covered vehicles' in the Cologne scenario. The x-axis indicates time, while the y-axis indicates the percentage of vehicles presenting at least one V2I contact opportunity.

6.5 Memoryless V2I Contacts Over Time

Now we present the performance of the deployment strategies over time. For each time window of 1 000s we show the percentage of vehicles experiencing at least one V2I contact opportunity. This experiment differs from the previous one (Section 6.5) in the sense that we treat time windows as memoryless (we do not remember vehicles covered on earlier time windows), and we consider just the active vehicles during the time window in order to compute the percentage of vehicles.

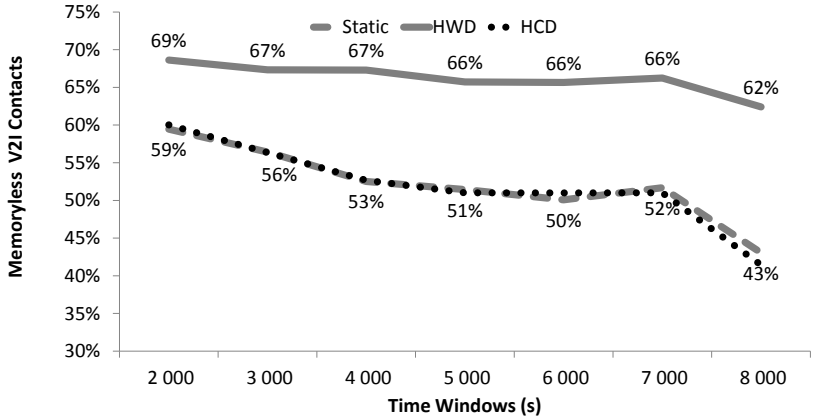


Figure 6.5. Memoryless Evaluation of V2I Contacts Over Time: The x-axis indicates time windows, while the y-axis indicates the percentage of active vehicles presenting at least one V2I contact opportunity.

Figure 6.5 presents our results considering Static Deployment, Hybrid Windowed Deployment, and Hybrid Cumulative Deployment. The x-axis indicates time windows,

while the y-axis indicates the percentage of active vehicles presenting at least one V2I contact opportunity. Now we notice that Hybrid Windowed Deployment overcomes Static Deployment and Hybrid Cumulative Deployment. Such result means that Hybrid Windowed Deployment recomputes the position of the roadside units in order to always maximize the number of distinct vehicles presenting contact opportunities inside a given time window. Thus, Hybrid Windowed Deployment is able to configure the roadside units in a layout offering maximum benefits at every instant of time. Just like the previous experiment, the performance of Hybrid Cumulative Deployment is very similar to Static Deployment.

6.6 Cumulative V2I Contacts Over Time

Now we present a cumulative view of the contact opportunities. For every instant of time we represent the total number of distinct vehicles presenting contact opportunities. Such number of vehicles are represented as a percentage of the total number of activated vehicles. In other words, we report the evolution of V2I contacts during the network operation. Static Deployment overcomes Hybrid Windowed Deployment for time $\geq$ 4 000s.

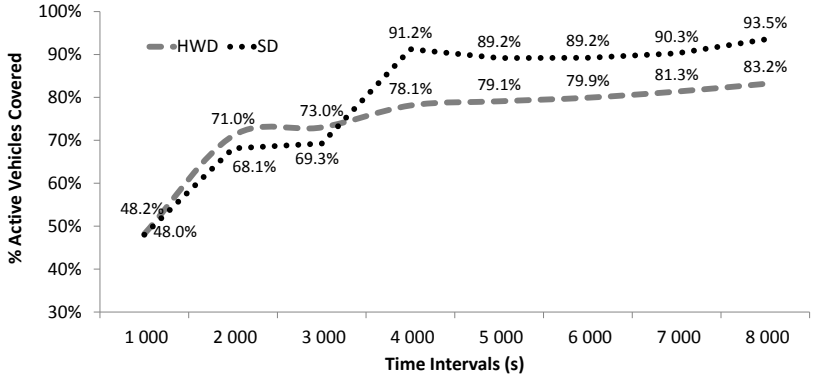


Figure 6.6. Deployment Performance During the Network Operation. The x-axis indicates time, while the y-axis indicates the number of distinct vehicles presenting contact opportunities as a percentage of the total number of activated vehicles.

Such results indicate the likelihood of a generic vehicle experiencing a V2I contact: for instance, a vehicle traveling during the time window 2 000s has a probability of 69% of driving-through a roadside unit. Because Hybrid Windowed Deployment is memoryless, we state that V2I contacts are random independent variables when we

consider distinct time windows. Thus, a vehicle traveling during time windows 2 000s and 3 000s has the following probabilities:

- Probability of 89.7% of experiencing at least one V2I contact during one time window;
- Probability of 46.2% of experiencing at least one V2I contact during both time windows;
- Probability of 10.3% of not experiencing any V2I contacts during both time windows.

Such determinism is not possible when we consider Static Deployment and Hybrid Cumulative Deployment because such strategies are not memoryless, and thus, the V2I encounters are not random independent variables.

6.7 Improvements of Hybrid Windowed Deployment over Static Deployment

In order to highlight the improvements of Hybrid Windowed Deployment over Static Deployment, we present Figure 6.7. In the worst-case analysis presented in Section 4.5.6, we reported that dynamic deployment would offer gains up to 60% in comparison to static deployment. The results presented here are in accordance with Section 4.5.6.

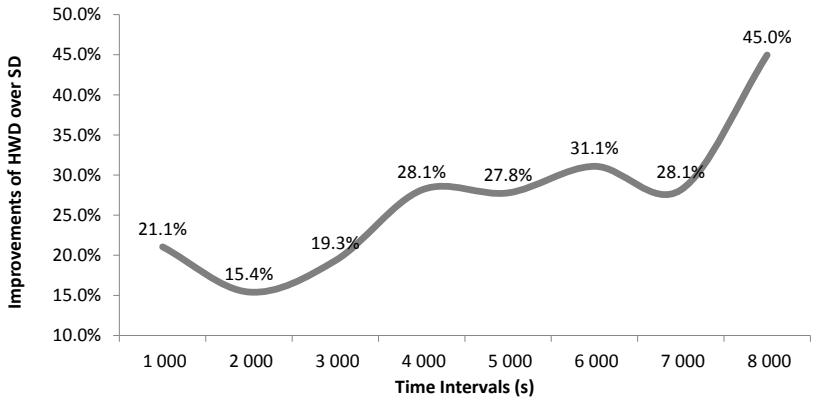


Figure 6.7. Improvements of Hybrid Windowed Deployment over Static Deployment.

6.8 Share of Stationary Roadside Units

Now we investigate the share of stationary and mobile roadside units composing our hybrid strategies. We split the Cologne scenario into time windows holding just 1 000s of traffic. Then, we run the deployment considering single windows. Finally, we compare pairs of adjacent time windows in order to reveal the number of roadside units preserving their original position. In Figure 6.8 we report our results. The x-axis indicates consecutive time windows, while the y-axis indicates the percentage of stationary roadside units of the deployment strategy.

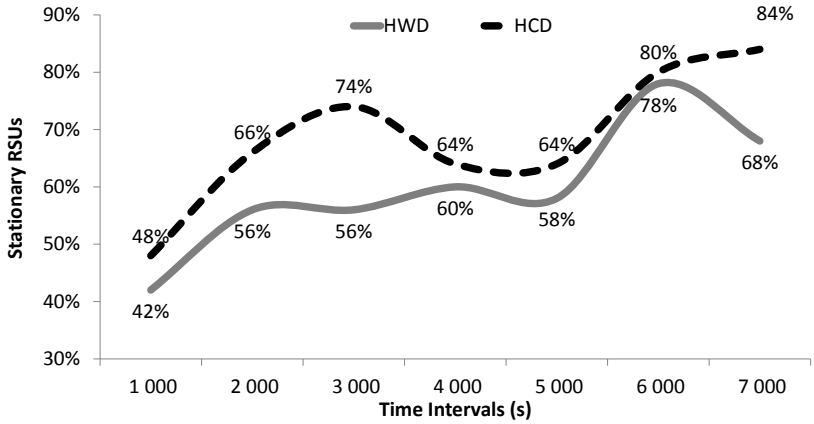


Figure 6.8. Percentage of Stationary RSUs in hybrid deployment: Figure presents the number of roadside units not changing their locations in consecutive time windows. The x-axis indicates consecutive time windows, while the y-axis indicates the percentage of stationary roadside units.

Hybrid Windowed Deployment is a more dynamic approach because the position of the roadside units are computed by considering just a single time window. Thus, Hybrid Windowed Deployment presents less roadside units keeping their old positions. Recall that Hybrid Cumulative Deployment considers all the previous time windows, and thus, it is less sensitive to traffic fluctuations.

Table 6.1. Percentage of Stationary Roadside Units.

Deployment	Median	Average	Std. dev.
Hybrid Windowed Deployment	58%	60%	11%
Hybrid Cumulative Deployment	66%	69%	12%

Table 6.1 summarizes the results. Because Hybrid Cumulative Deployment does not offer relevant gains over the Static Deployment (Sections 6.5 and 6.6, it seems somehow pointless to move the roadside units using the Hybrid Cumulative Deployment strategy.

6.9 Absolute V2I Contact Time

Figure 6.9 presents the V2I contact time for the three strategies. We notice that Hybrid Windowed Deployment offers higher V2I contact time, after all, Hybrid Windowed Deployment always maximizes the number of V2I contacts within a given time window. When we also consider that Hybrid Windowed Deployment covers less distinct vehicles (results of Section 6.5), we conclude that Hybrid Windowed Deployment presents vehicles covered on more than one time window. Table 6.2 summarizes these results.

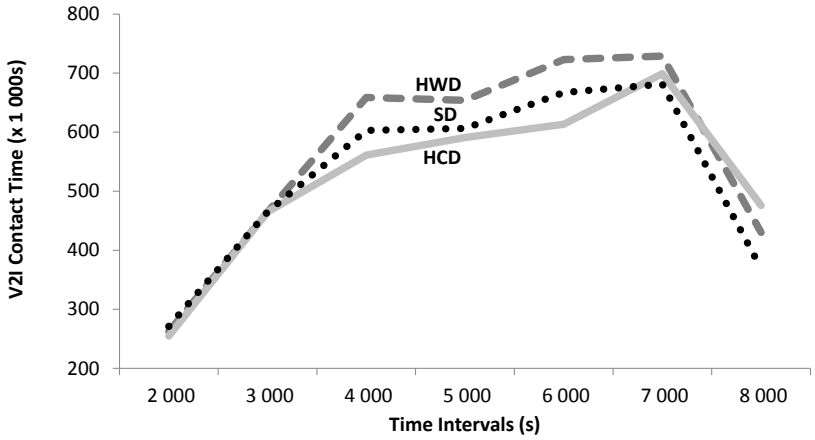


Figure 6.9. V2I Contact Time: Figure indicates V2I contact time for HWD, SD and HCD. The x-axis indicates the time window, while the y-axis indicates the contact time (x1 000s).

Table 6.2. Absolute V2I Contact Time.

Deployment	Median (x1 000s)	Average (x1 000s)	Std. dev. (x1 000s)
Hybrid Windowed Deployment	559	494	248
Hybrid Cumulative Deployment	518	461	218
Static Deployment	534	461	227

6.10 Relative V2I Contact Time

In order to highlight the result presented in Figure 6.9 we propose Figure 6.9. We indicate the V2I contact time as a percentage of maximum contact time achieved for a given time window (presented in Equation 6.1). The x-axis indicates the time window,

while the y-axis indicates contact time (as a percentage of maximum contact time achieved in the given time window).

$$p_t^d = \frac{c_t^d}{max(c_t^k)\ \forall k \in \{SD, HWD, HCD\}} \tag{6.1}$$

Where:

p_t^d= Percentage of contacts in deployment d at window t.

c_t^d= Number of contacts in deployment d at window t.

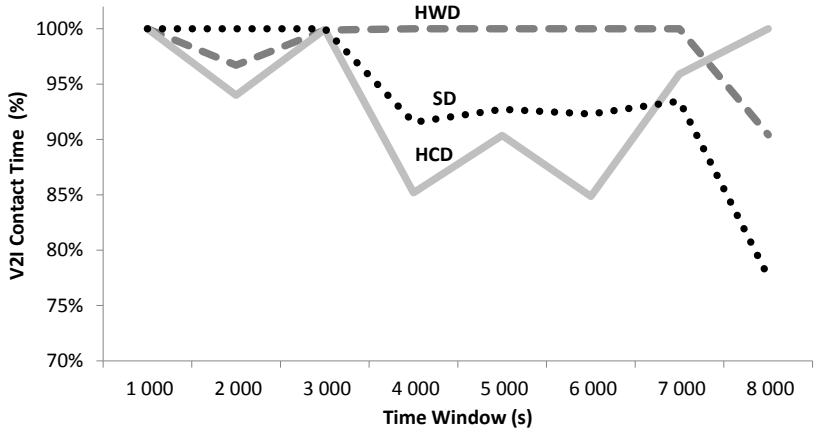


Figure 6.10. V2I Contact Time: Figure indicates contact time as a percentage of the maximum contact time achieved for a given time window.

Hybrid Windowed Deployment presents higher contact time. Because Hybrid Windowed Deployment only accounts for the mobility within the time window, it presents higher contribution during most of the experiment.

Table 6.3 summarizes these results.

Table 6.3. Relative V2I Contact Time.

Deployment	Median	Average	Std. dev.
Hybrid Windowed Deployment	100%	98%	3%
Hybrid Cumulative Deployment	95%	94%	6%
Static Deployment	93%	93%	7%

6.11 Displacement of Mobile Roadside Units

Figure 6.11 presents traveled distance over time for mobile roadside units considering Hybrid Windowed Deployment and Hybrid Cumulative Deployment. Hybrid

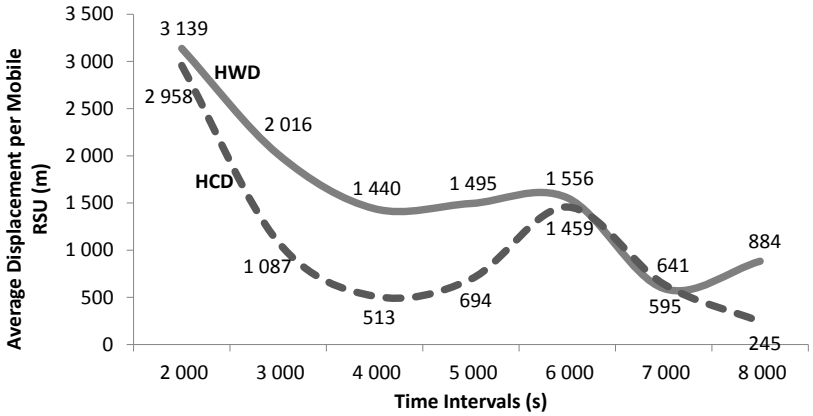


Figure 6.11. Average Displacement per RSU in meters.

Windowed Deployment is a more dynamic strategy, thus it more changes to the infrastructure topology.

Given a time window of 1 000s, and a median displacement of 1 495m, the mobile roadside units must travel a median of 5.2km/h, a reasonable speed able to be achieved by public transportation system.

Table 6.4 summarizes these results.

Table 6.4. Characterizing the Mobility of RSUs.

Deployment	Med.Dist. (km)	Avg.Dist. (km)	Std. dev. (km)	Med.Speed (km/h)	Max.Speed (km/h)
HWD	1.5	1.6	0.8	5.2	11.3
HCD	0.7	1.1	0.9	2.5	10.6

6.12 Duration of Contact Opportunities per Vehicle/RSU

The type of physical communication to be adopted by the vehicular network depends on the duration of contact opportunities. In Figure 6.12 the x-axis presents the time windows for our sample, while the y-axis indicates the average contact duration for each vehicle crossing a roadside unit. Average contact time is 27.6s in Static Deployment and 26.02s in HWD. Minimum contact time is 11.83s (Static) and 10.34s, while maximum contact time is 215.80s (Static) and 218.15s (HWD). Table 6.5 summarizes these results.

The duration of the contact opportunities does not change when we select Static Deployment or Hybrid Windowed Deployment. In fact, this study presents

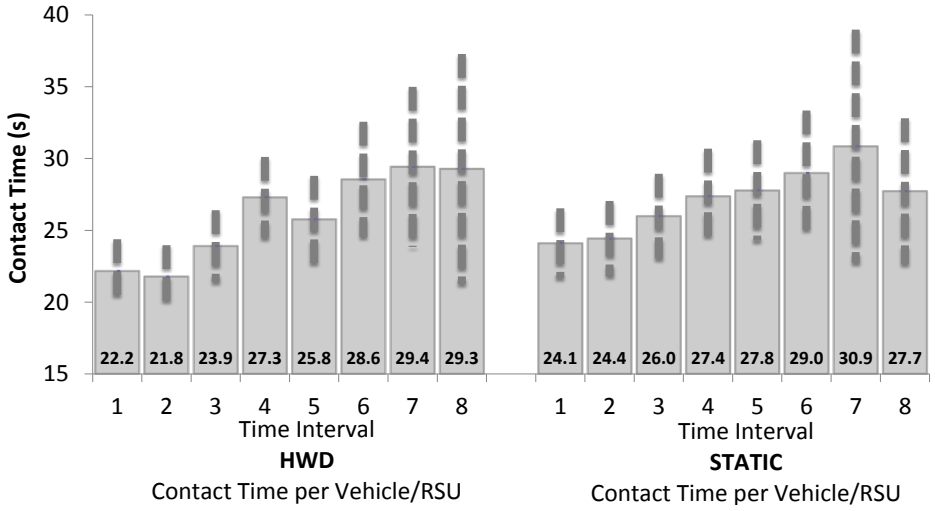


Figure 6.12. Duration of Contact Opportunities.

the duration of contact opportunities assuming the physical proximity as a unique decision parameter. On the other hand, we have partitioned the Cologne scenario in a 100x100-grid in order to achieve ranges of coverage of approximately 230m, a typical range assumed to present high contact probability: Teixeira et al. (2014) present a field test evaluation of the IEEE 802.11p standard. Authors report an average bit rate of 3Mbps when considering 500 bytes UDP packets, and 8Mbps for 1460 bytes packets. Packet loss is around 1% for distances below 300m, and association time is 1.035s. Thus, our results somehow reflect the expected contact time in a real scenario. Complementary, when we consider the duration of contact opportunities and the time required for establishing new connections, we trivially conclude that very small urban cells may degenerate the network performance: very small urban cells imply in very short contact times, and the network would spend a large effort in establishing connections, and just a portion of the communication would be really useful. Obviously, we can also work on the development of strategies to decrease the required time for establishing new connections.

Table 6.5. Duration of Contact Opportunities.

Measure	Static (s)	HWD (s)
Average	27.16	26.02
Median	23.43	22.99
Std. Dev.	15.90	15.46
Minimum	11.83	10.34
Maximum	215.80	218.15
Confidence (95%)	1.56	1.52

6.13 Analysis

We investigate the potential use of hybrid architectures composed of stationary and mobile roadside units. We evaluate three distinct deployment strategies: (i) Stationary Deployment: Algorithm evaluates the entire mobility trace, and then it locates the stationary roadside units; (ii) Hybrid Windowed Deployment (HWD): Algorithm evaluates the position of each roadside unit periodically. We have arbitrarily defined a time window of 1 000s; (iii) Hybrid Cumulative Deployment (HCD): Algorithm also evaluates the position of each roadside unit periodically, but the time windows extends from the beginning of the mobility trace up to the actual moment. We have arbitrarily defined time windows of 1 000s.

Our hybrid architecture models the deployment of roadside units as a Maximum Coverage Problem, and we solve the deployment using the well-know greedy solution. Given the complexity of the Cologne scenario, we propose partitioning the road network into same size cells. Our results demonstrate that a 100x100 grid represents a good compromise between processing efforts and deployment accuracy.

Road network partitioning has been already studied in the vehicular networks literature. However, authors typically apply the partitioning as an intermediate to step to model the problem. Instead of it, we visualize the partition as a very wise and economic alternative to model a road network and its associated flows.

Our main findings are:

- Number of distinct vehicles contacting the infrastructure is maximized in Static Deployment (93.5%), while Hybrid Windowed Deployment is 83.2%;
- Number of vehicles contacting the infrastructure is maximized in Hybrid Windowed Deployment (69%), while Static Deployment is 59%;
- Hybrid Windowed Deployment may improve Static Deployment up to 45%, depending on the time window under analysis;
- During adjacent time windows, 58% of the roadside units deployed by Hybrid Windowed Deployment preserves their positions;
- In order to implement Hybrid Windowed Deployment, the 42% mobile roadside units must travel at speeds ranging from of 5.2km/h up to 11.3km/h;
- Hybrid Cumulative Deployment offers performance indicators very close to Static Deployment. Thus, we do not recommend such strategy.

Chapter 7

Final Remarks

Throughout this thesis we have developed and evaluated the use of partial mobility information assisting the deployment of roadside units to support the operation of vehicular networks. During this work we have conducted an in-depth review of 11 years of literature addressing infrastructure-based vehicular networks. We could identify 30 distinct strategies for the deployment of infrastructure for vehicular networks. Most of the efforts consider unrealistic topologies and/or unrealistic mobility, which possibly implies in distortion on reported results. The allocation of roadside units within an urban area is an optimization problem by its nature, and we had to select an objective function to be target by our deployment. There are several possibilities, and we have chosen to maximize the number of distinct vehicles contacting the infrastructure at least once during the trip, an interesting metric when we intend to collect and disseminate traffic announcements. We have considered three deployment strategies as our baseline:

- MCP-g: Heuristic based on the greedy solution for the Maximum Coverage Problem. At each step the heuristic selects the urban cell presenting higher number of uncovered vehicles. In order to solve the deployment, the heuristic requires previous knowledge of vehicles trajectories;

- MCP-based: Heuristic based on an approximate solution for the Maximum Coverage Problem. At each step the heuristic iteratively selects the urban cell presenting higher number of vehicles;

- Opt_{dv}: Integer Linear Programming Formulation for the deployment based on Maximum Coverage Problem.

We have proposed three deployment strategies:

- PMCP-based: Heuristic based on an approximate solution for the Probabilistic Maximum Coverage Problem. At each step the heuristic iteratively selects the urban cell presenting higher flow projection;
- FPF: Heuristic extends the projection of the flow to the entire road network;
- Hybrid Windowed Deployment (HWD): Deployment strategy proposing the use of a hybrid architecture based on stationary and mobile roadside units. The heuristic analyzes instantaneous time intervals in order to propose sets of stationary and mobile roadside units;
- Hybrid Cumulative Deployment (HCD): Deployment strategy proposing the use of a hybrid architecture based on stationary and mobile roadside units. The heuristic analyzes time cumulative time intervals in order to propose sets of stationary and mobile roadside units.

Several deployment strategies consider placing the roadside units in densest places. Although placing the roadside units in densest places may seem reasonable at a first glance, the assumption fails when we consider that vehicles composing the dense regions are originated from nearby, and the dense region is a result of merging flows. In other words, dense regions do not appear as isolated islands, but they are the result of mixed flows converging to a given location. Such issue indicates dense regions tend to appear somehow interconnected, and vehicles traveling nearby dense areas have high probability of joining the main flow. Thus, when we consider just the concentration of vehicles to allocate the communication infrastructure, we may incur in redundant coverage, i.e., we may deploy roadside units covering the same flow in distinct locations. In order to eliminate the redundant coverage we propose the adoption of a probabilistic approach able to describe the vehicular mobility. The core of our approach is a stochastic matrix defining vehicles' migration ratios. By using stochastic matrix we have formulated the allocation of roadside units as a Probabilistic Maximum Coverage Problem.

Probabilistic Maximum Coverage Problem can be informally stated as a collection of sets defined over a domain of elements in which each one of these elements have a probability of belonging to a given set. Our goal is to select those α sets resulting in maximum expected cardinality. Our initial application of the probabilistic coverage to solve the allocation of roadside units was carried out on a theoretical perfect grid road network. The decision to adopt the grid-based road network has an important role in the development of our strategy: its symmetry has allowed us to understand key aspects of the probabilistic approach. Our results have demonstrated that our

PMCP-based approach distributes the roadside units in a layout better fitting the traffic flow. Furthermore, the projection of the flow has an interesting property: it allows us to identity interconnected dense areas. The projection of the flow can be roughly imagined as distributing parts of each vehicle over the entire road network. In the next step we select the location presenting higher expectation of receiving vehicles, and then we proceed to pullout projection when we remove all 'covered' parts from the entire road network. After the pullout projection, all we have left in the road network are 'uncovered' items. Thus, it is somehow intuitive our strategy is supposed to always incur in less redundant coverage than traditional placement of roadside units considering just the concentration of vehicles.

After the development and initial evaluation of the probabilistic approach we have searched for possible alternatives to adapt our strategy to real road networks. But, instead of adapting our strategy to address arbitrary topologies, we have chosen to adapt arbitrary topologies to fit our strategy. In practical terms, we have devised a strategy able to convert any possible road network into a grid-based road network. We propose partitioning the road network, i.e., divide the urban area in a set of adjacent same-size cells. Once the road network is partitioned we discard the (complex) road network and we manipulate the flow between adjacent grid cells in a divide-and-conquer strategy. Using the resulting urban cells, we apply the probabilistic model to select those better candidates to receive the roadside units.

Then, we have proposed a set of experiments to evaluate the partitioning (and also to reevaluate the probabilistic deployment), but now using a real road network with randomized flow of vehicles. The use of a randomized flow of vehicles is intentional because it offers a worst-case analysis of the deployment performance. Our main finding is: while traffic changes over time, we may rely on dynamic methods to improve the resources allocation: when we recompute the position of each roadside unit every 1 000s (for example), we improve the vehicle-to-infrastructure contact time in almost 60%. Next, we evaluate our deployment algorithm considering a real road network and a realistic vehicular trace. The results demonstrate 89.6% of the vehicles experience at least one V2I contact by covering just 2.5% of the entire road network. Now we compare PMCP-based to MCP-based and also MCP-g (greedy solution for MCP). Our main finding is our PMCP-based approach presents a clear lower bound offering 'minimum guarantees' of efficiency for network providers. Such guarantees are not present in MCP-based nor MCP-g.

Finally, we investigate an architecture composed of stationary and mobile roadside units. It is a common sense that traffic fluctuates: thus, an architecture employing just stationary roadside units might not be able to properly support the

network operation all the time. Similarly, an architecture composed just of mobile roadside units makes a few sense when we consider that the road network does not change that often. In other words, traffic fluctuations are limited by its subjacent road network. As major roads counts on a higher transportation capacity, they tend to be very popular routes, and they are natural candidates for receiving stationary roadside units. And mobile roadside are highlighted as ideal solutions to handle in-borders traffic. In order to improve the deployment performance, we can rely on hybrid deployment strategies employing sets of stationary and mobile roadside units. Stationary roadside units act as a main backbone for data dissemination covering the most important regions of an urban area (i.e., regions known as always presenting relevant traffic). On the other hand, we can rely on mobile roadside units to address traffic variations. Our results demonstrate that hybrid deployment increases the number of V2I contacts up to 45%.

Now we return to our intended contribution presented in Chapter 1:

> **Design of a deployment algorithm employing a new paradigm of mobility information.**

This thesis demonstrates (in a step-by-step basis) the process of building and evaluating a new deployment algorithm addressing our contribution. Our proposal uses a probabilistic approach to infer the mobility of vehicles along the road network in order to compute the location of the roadside units, which seems to be an interesting contribution given the high mobility of vehicles. The only required information are vehicles' migration ratios.

We demonstrate that previously knowledge of the vehicles trajectories is not mandatory for achieving a close-to-optimal deployment performance: by covering 1.0% of the Cologne's road network, our FPF deployment yields 89.8% of all vehicles experiencing at least one V2I contact opportunity. On the other hand, if we assume previous knowledge of individual vehicles trajectories, we improve the performance in just 2.3%.

7.1 Future Works

Throughout this work we have focused on the delivery of small traffic announcements. But a vehicular network is supposed to route more than traffic announcements. Thus, studies are also necessary to evaluate deployment strategies for other types of content. Streaming and real-time data delivery are highlighted as promising dissemination

paradigms for the vehicular environment. Although we demonstrate the viability of hybrid deployment, more research is still required to investigate, for instance, the possibility of integrating vehicles from the public transportation system to serve as mobile roadside units. Hybrid deployment may also be investigated to determine layouts of roadside units offering maximum off-line constraints. Such strategy would serve as a basis for the dissemination of real-time traffic. Similarly, we also envision that partial mobility information would improve the performance of other deployment strategies.

Although the Integer Linear Programming Formulation has demonstrated low scalability, it raises the opportunity for the development of meta-heuristics designed specifically to improve the performance of the model. Finally, our results demonstrate that our PMCP-based approach provides layouts of roadside units more similar to MCP-based, than MCP-g. In order to improve PMCP-based we envision a 2-levels road network partitioning. First level is responsible for defining the number of roadside units covering each macro region. Second level applies the PMCP-based approach. Our goal is to increase the distribution of roadside units within the road network.

We also highlight the importance of running traces of real applications in order to identify the most suitable applications for our deployment scenario. The use of LTE must also be investigated through the proposal of offloading strategies.

-x-x-

Bibliography

Abdrabou, A., Liang, B., and Zhuang, W. (2010). Delay analysis for a reliable message delivery in sparse vehicular ad hoc networks. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–5. IEEE. ISSN 1930-529X.

Abdrabou, A. and Zhuang, W. (2009). On a stochastic delay bound for disrupted vehicle-to-infrastructure communication with random traffic. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1–6. IEEE. ISSN 1930-529X.

Abdrabou, A. and Zhuang, W. (2011). Probabilistic delay control and road side unit placement for vehicular ad hoc networks with disrupted connectivity. *Selected Areas in Communications, IEEE Journal on*, 29(1):129–139. ISSN 0733-8716.

Akyildiz, I. F., Wang, X., and Wang, W. (2005). Wireless mesh networks: a survey. *Computer Networks*, 47(4):445 – 487. ISSN 1389-1286.

Amir, Y., Danilov, C., Hilsdale, M., Musăloiu-Elefteri, R., and Rivera, N. (2006). Fast handoff for seamless wireless mesh networks. In *Proceedings of the 4th International Conference on Mobile Systems, Applications and Services*, MobiSys '06, pages 83--95, New York, NY, USA. ACM.

Annese, S., Casetti, C., Chiasserini, C., Di Maio, N., Ghittino, A., and Reineri, M. (2011). Seamless connectivity and routing in vehicular networks with infrastructure. *Selected Areas in Communications, IEEE Journal on*, 29(3):501–514. ISSN 0733-8716.

Araujo, R., Igreja, A., de Castro, R., and Araujo, R. (2012). Driving coach: A smartphone application to evaluate driving efficient patterns. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 1005--1010. IEEE, IEEE.

Aslam, B., Amjad, F., and Zou, C. (2012). Optimal roadside units placement in urban areas for vehicular networks. In *Computers and Communications (ISCC), 2012 IEEE Symposium on*, pages 000423–000429. IEEE. ISSN 1530-1346.

Aurenhammer, F. (1991). Voronoi diagrams—a survey of a fundamental geometric data structure. *ACM Comput. Surv.*, 23(3):345--405. ISSN 0360-0300.

Banerjee, N., Corner, M. D., Towsley, D., and Levine, B. N. (2008). Relays, base stations, and meshes: Enhancing mobile networks with infrastructure. In *Proceedings of the 14th ACM International Conference on Mobile Computing and Networking*, MobiCom '08, pages 81--91, New York, NY, USA. ACM.

Banzhaf, W., Nordin, P., Keller, R. E., and Francone, F. D. (1998). *Genetic programming: an introduction*, volume 1. Morgan Kaufmann San Francisco.

Bonabeau, E., Dorigo, M., and Theraulaz, G. (1999). *Swarm intelligence: from natural to artificial systems*. Number 1. Oxford university press.

Borsetti, D. and Gozalvez, J. (2010). Infrastructure-assisted geo-routing for cooperative vehicular networks. In *Vehicular Networking Conference (VNC), 2010 IEEE*, pages 255–262. ISSN 2157-9857.

Boukerche, A., Oliveira, H. A., Nakamura, E. F., and Loureiro, A. A. (2008). Vehicular ad hoc networks: A new challenge for localization-based systems. *Computer Communications*, 31(12):2838 – 2849. ISSN 0140-3664. Mobility Protocols for ITS/VANET.

Brik, V., Mishra, A., and Banerjee, S. (2005). Eliminating handoff latencies in 802.11 wlans using multiple radios: Applications, experience, and evaluation. In *Proceedings of the 5th ACM SIGCOMM Conference on Internet Measurement*, IMC '05, pages 27--27, Berkeley, CA, USA. USENIX Association.

Bruno, R. and Nurchis, M. (2013). Robust and efficient data collection schemes for vehicular multimedia sensor networks. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a*, pages 1–10.

Bychkovsky, V., Chen, K., Goraczko, M., Hu, H., Hull, B., Miu, A., Shih, E., Zhang, Y., Balakrishnan, H., and Madden, S. (2006). The cartel mobile sensor computing system. In *In Proceedings of the 4th international conference on Embedded networked sensor systems, SenSys*, page 383–384. ACM.

Cai, C., Wang, Y., and Geers, G. (2010). Adaptive traffic signal control using vehicle-to-infrastructure communication: a technical note. In *Proceedings of the Second International Workshop on Computational Transportation Science*, pages 43--47. ACM.

Capone, A., Cesana, M., Napoli, S., and Pollastro, A. (2007). Mobimesh: a complete solution for wireless mesh networking. In *Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE Internatonal Conference on*, pages 1–3.

Cataldi, P. and Harri, J. (2011). User/operator utility-based infrastructure deployment strategies for vehicular networks. In *Vehicular Technology Conference (VTC Fall), 2011 IEEE*, pages 1–5. ISSN 1090-3038.

Cavalcante, E. S., Aquino, A. L., Pappa, G. L., and Loureiro, A. A. (2012). Roadside unit deployment for information dissemination in a vanet: An evolutionary approach. In *Proceedings of the Fourteenth International Conference on Genetic and Evolutionary Computation Conference Companion*, GECCO Companion '12, pages 27--34, New York, NY, USA. ACM.

Cheng, H., Fei, X., Boukerche, A., Mammeri, A., and Almulla, M. (2013). A geometry-based coverage strategy over urban vanets. In *Proceedings of the 10th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks*, PE-WASUN '13, pages 121--128, New York, NY, USA. ACM.

Chi, J., Jo, Y., Park, H., and Park, S. (2013). Intersection-priority based optimal rsu allocation for vanet. In *Ubiquitous and Future Networks (ICUFN), 2013 Fifth International Conference on*, pages 350–355. ISSN 2165-8528.

Chiu, S. and Chand, S. (1993). Adaptive traffic signal control using fuzzy logic. In *Fuzzy Systems, 1993., Second IEEE International Conference on*, pages 1371--1376. IEEE.

Chun, S. and Mao, G. (2010). Analysis of k-hop connectivity probability in 2-d wireless networks with infrastructure support. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–5. ISSN 1930-529X.

Chun, S., Zhang, W., Yang, Y., and Mao, G. (2010). Analysis of access and connectivity probabilities in infrastructure-based vehicular relay networks. In *Wireless Communications and Networking Conference (WCNC), 2010 IEEE*, pages 1–6. ISSN 1525-3511.

Chung, K. L. (1967). *Markov chains*. Springer.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2001). *Introduction to algorithms*. MIT press.

Dantzig, G. B. (1998). *Linear programming and extensions*. Princeton university press.

Eriksson, J., Girod, L., Hull, B., Newton, R., Madden, S., and Balakrishnan, H. (2008). The pothole patrol: using a mobile sensor network for road surface monitoring. In *ACM MobiSys*.

Fan, X. and Li, V.-K. (2011). The probabilistic maximum coverage problem in social networks. In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pages 1–5. ISSN 1930-529X.

Fernandez Ruiz, P., Nieto Guerra, C., and Skarmeta, A. (2010). Deployment of a secure wireless infrastructure oriented to vehicular networks. In *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*, pages 1108–1114. ISSN 1550-445X.

Filippini, I., Malandrino, F., Dan, G., Cesana, M., Casetti, C., and Marsh, I. (2012). Non-cooperative rsu deployment in vehicular networks. In *Wireless On-demand Network Systems and Services (WONS), 2012 9th Annual Conference on*, pages 79–82.

Fiore, M. and Härri, J. (2008). The networking shape of vehicular mobility. In *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '08, pages 261--272, New York, NY, USA. ACM.

Foy, M. D., Benekohal, R. F., and Goldberg, D. E. (1992). Signal timing determination using genetic algorithms. *Transportation Research Record*, (1365).

Franz, W., Eberhardt, R., and Luckenbach, T. (2001). Fleetnet-internet on the road. In *8th World Congress on Intelligent Transport Systems*.

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

Gakenheimer, R. (1999). Urban mobility in the developing world. *Transportation Research Part A: Policy and Practice*, 33(7–8):671 – 689. ISSN 0965-8564.

García-Nieto, J., Alba, E., and Olivera, A. C. (2012). Swarm intelligence for traffic light scheduling: Application to real urban areas. *Engineering Applications of Artificial Intelligence*, 25(2):274 – 283. ISSN 0952-1976. Special Section: Local Search Algorithms for Real-World Scheduling and Planning.

Gass, S. I. (1958). *Linear programming*. Wiley Online Library.

Gerla, M. and Kleinrock, L. (2011). Vehicular networks and the future of the mobile internet. *Computer Networks*, 55(2):457 – 469. ISSN 1389-1286. Wireless for the Future Internet.

Gerla, M., Zhou, B., Lee, Y.-Z., Soldo, F., Lee, U., and Marfia, G. (2006). Vehicular grid communications: the role of the internet infrastructure. In *Proceedings of the 2nd annual international workshop on Wireless internet*, page 19. ACM.

Gerlough, D. L. and Schuhl, A. (1955). *Use of Poisson distribution in highway traffic*. Eno Foundation for Highway Traffic Control.

Gibbons, R. (1992). *A primer in game theory*. Harvester Wheatsheaf New York.

Habib, S. and Safar, M. (2007). Sensitivity study of sensors' coverage within wireless sensor networks. In *Proceedings of 16th International Conference on Computer Communications and Networks (ICCCN)*, pages 876--881. IEEE.

Hadaller, D., Keshav, S., and Brecht, T. (2006). Mv-max: improving wireless infrastructure access for multi-vehicular communication. *Proceeding of the 2006 SIGCOMM workshop*, pages 269--276.

Hamza-Lup, G., Hua, K., Le, M., and Peng, R. (2008). Dynamic plan generation and real-time management techniques for traffic evacuation. *Intelligent Transportation Systems, IEEE Transactions on*, 9(4):615–624. ISSN 1524-9050.

Harigovindan, V., Babu, A., and Jacob, L. (2014). Proportional fair resource allocation in vehicle-to-infrastructure networks for drive-thru internet applications. *Computer Communications*, 40(0):33 – 50. ISSN 0140-3664.

Harsanyi, J. C. and Selten, R. (1988). *A general theory of equilibrium selection in games*, volume 1. The MIT Press.

Hartenstein, H. and Laberteaux, K. (2008). A tutorial survey on vehicular ad hoc networks. *Communications Magazine, IEEE*, 46(6):164–171. ISSN 0163-6804.

Hsieh, H.-Y. and Sivakumar, R. (2002). Internetworking wwans and wlans in next generation wireless data networks. In *Proceedings of the International Conference on 3G Wireless and Beyond*. Citeseer.

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

Jerbi, M., Senouci, S., Doudane, Y., and Beylot, A. L. (2008). Geo-localized virtual infrastructure for urban vehicular networks. In *ITS Telecommunications, 2008. ITST 2008. 8th International Conference on*, pages 305–310.

Johnson, D. A. and Trivedi, M. M. (2011). Driving style recognition using a smartphone as a sensor platform. In *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*, pages 1609--1615. IEEE.

Kaelbling, L. P., Littman, M. L., and Moore, A. W. (1996). Reinforcement learning: A survey. *arXiv preprint cs/9605103*.

Karakuzu, C. and Demirci, O. (2010). Fuzzy logic based smart traffic light simulator design and hardware implementation. *Applied Soft Computing*, 10(1):66 – 73. ISSN 1568-4946.

Kaufman, L. and Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis*, volume 344. John Wiley & Sons.

Kawashima, H. (1990). Japanese perspective of driver information systems. *Transportation*, 17(3):263–284. ISSN 0049-4488.

Kchiche, A. and Kamoun, F. (2010). Centrality-based access-points deployment for vehicular networks. In *Telecommunications (ICT), 2010 IEEE 17th International Conference on*, pages 700–706.

Kemeny, J. G. and Snell, J. L. (1960). *Finite markov chains*, volume 356. van Nostrand Princeton, NJ.

Keränen, A., Ott, J., and Kärkkäinen, T. (2009). The ONE Simulator for DTN Protocol Evaluation. In *SIMUTools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, New York, NY, USA. ICST.

Khabazian, M. and Mehmet Ali, M. (2007). Generalized performance modeling of vehicular ad hoc networks (vanets). In *Computers and Communications, 2007. ISCC 2007. 12th IEEE Symposium on*, pages 51–56. ISSN 1530-1346.

Kitchenham, B. (2004). Procedures for performing systematic reviews. *Technical Report TR/SE -0401, Software Engineering Group, Department of Computer Science, Keele University, UK.*

Kitchenham, B. A. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering, version 2.3. *Technical Report EBSE 2007-01, Software Engineering Group, School of Computer Science and Mathematics, Keele University and Department of Computer Science University of Durham.*

Koegel, M., Kiess, W., Kerper, M., and Mauve, M. (2011). Compact vehicular trajectory encoding. In *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, pages 1–5. ISSN 1550-2252.

Korkmaz, G., Ekici, E., and Ozguner, F. (2006). A cross-layer multihop data delivery protocol with fairness guarantees for vehicular networks. *Vehicular Technology, IEEE Transactions on*, 55(3):865–875. ISSN 0018-9545.

Korkmaz, G., Ekici, E., and Özgüner, F. (2010). Supporting real-time traffic in multihop vehicle-to-infrastructure networks. *Transportation Research Part C: Emerging Technologies*, 18(3):376 – 392. ISSN 0968-090X. 11th {IFAC} Symposium: The Role of Control.

Koukoumidis, E., Peh, L.-S., and Martonosi, M. R. (2011). Signalguru: leveraging mobile phones for collaborative traffic signal schedule advisory. In *Proceedings of the 9th international conference on Mobile systems, applications, and services*, pages 127--140. ACM.

Koza, J. R. (1992). *Genetic programming: on the programming of computers by means of natural selection*, volume 1. MIT press.

Kozat, U. C. and Tassiulas, L. (2003). Throughput capacity of random ad hoc networks with infrastructure support. In *Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 55--65. ACM.

Larson, H. (1982). Introduction to probability theory and statistical inference.

Le, T., Cai, C., and Walsh, T. (2011). Adaptive signal-vehicle cooperative controlling system. In *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*, pages 236--241. IEEE.

Lee, J. and Kim, C. (2010). A roadside unit placement scheme for vehicular telematics networks. In Kim, T.-h. and Adeli, H., editors, *Advances in Computer Science and*

Information Technology, volume 6059 of *Lecture Notes in Computer Science*, pages 196–202. Springer Berlin Heidelberg.

Li, P., Huang, X., Fang, Y., and Lin, P. (2007). Optimal placement of gateways in vehicular networks. *Vehicular Technology, IEEE Transactions on*, 56(6):3421–3430. ISSN 0018-9545.

Liang, H. and Zhuang, W. (2012). Cooperative data dissemination via roadside wlans. *Communications Magazine, IEEE*, 50(4):68–74. ISSN 0163-6804.

Liang, Y., Liu, H., and Rajan, D. (2012). Optimal placement and configuration of roadside units in vehicular networks. In *Vehicular Technology Conference (VTC Spring), 2012 IEEE 75th*, pages 1–6. ISSN 1550-2252.

Liu, K. and Lee, V.-S. (2010). Rsu-based real-time data access in dynamic vehicular networks. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 1051–1056. ISSN 2153-0009.

Liu, Y., Niu, J., Ma, J., and Wang, W. (2013). File downloading oriented roadside units deployment for vehicular networks. *Journal of Systems Architecture*, 59(10, Part B):938 – 946. ISSN 1383-7621. Advanced Smart Vehicular Communication System and Applications.

Liu, Z. (2007). A survey of intelligence methods in urban traffic signal control. *IJCSNS International Journal of Computer Science and Network Security*, 7(7):105--112.

Liya, X., Chuanhe, H., Peng, L., and Junyu, Z. (2013). A randomized algorithm for roadside units placement in vehicular ad hoc network. In *Mobile Ad-hoc and Sensor Networks (MSN), 2013 IEEE Ninth International Conference on*, pages 193–197.

Lochert, C., Scheuermann, B., Caliskan, M., and Mauve, M. (2007). The feasibility of information dissemination in vehicular ad-hoc networks. In *Wireless on Demand Network Systems and Services, 2007. WONS '07. Fourth Annual Conference on*, pages 92–99.

Lochert, C., Scheuermann, B., Wewetzer, C., Luebke, A., and Mauve, M. (2008). Data aggregation and roadside unit placement for a vanet traffic information system. In *Proceedings of the Fifth ACM International Workshop on VehiculAr Inter-NETworking*, VANET '08, pages 58--65, New York, NY, USA. ACM.

Lu, N., Zhang, N., Cheng, N., Shen, X., Mark, J., and Bai, F. (2013). Vehicles meet infrastructure: Toward capacity-cost tradeoffs for vehicular access networks.

Intelligent Transportation Systems, IEEE Transactions on, 14(3):1266–1277. ISSN 1524-9050.

Luan, T., Cai, L., Chen, J., Shen, X., and Bai, F. (2013). Engineering a distributed infrastructure for large-scale cost-effective content dissemination over urban vehicular networks. *Vehicular Technology, IEEE Transactions on*, PP(99):1–1. ISSN 0018-9545.

Luan, T. H., Ling, X., and Shen, X. S. (2012). Provisioning qos controlled media access in vehicular to infrastructure communications. *Ad Hoc Networks*, 10(2):231 – 242. ISSN 1570-8705. Recent Advances in Analysis and Deployment of {IEEE} 802.11e and {IEEE} 802.11p Protocol Families.

Luenberger, D. G. (1973). *Introduction to linear and nonlinear programming*, volume 28. Addison-Wesley Reading, MA.

Luo, J., Gu, X., Zhao, T., and Yan, W. (2010). Mi-vanet: A new mobile infrastructure based vanet architecture for urban environment. In *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, pages 1–5. ISSN 1090-3038.

Macedo, D. F., de Oliveira, S., Teixeira, F. A., Aquino, A. L. L., and Rabelo, R. A. (2012). (cia)2-its: Interconnecting mobile and ubiquitous devices for intelligent transportation systems. In *Work in Progress session at PerCom 2012*.

Malandrino, F., Casetti, C., Chiasserini, C., and Fiore, M. (2011). Content downloading in vehicular networks: What really matters. In *INFOCOM, 2011 Proceedings IEEE*, pages 426–430. ISSN 0743-166X.

Mangel, T. and Hartenstein, H. (2011). An analysis of data traffic in cellular networks caused by inter-vehicle communication at intersections. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 473–478. ISSN 1931-0587.

Marfia, G., Pau, G., De Sena, E., Giordano, E., and Gerla, M. (2007). Evaluating vehicle network strategies for downtown portland: opportunistic infrastructure and the importance of realistic mobility models. In *Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*, pages 47--51. ACM.

Markov, A. (1971). Extension of the limit theorems of probability theory to a sum of variables connected in a chain.

Marmol, F. G. and Perez, G. M. (2012). Trip, a trust and reputation infrastructure-based proposal for vehicular ad hoc networks. *Journal of Network*

and Computer Applications, 35(3):934 – 941. ISSN 1084-8045. Special Issue on Trusted Computing and Communications.

McKenney, D. and White, T. (2013). Distributed and adaptive traffic signal control within a realistic traffic simulation. *Engineering Applications of Artificial Intelligence*, 26(1):574 – 583. ISSN 0952-1976.

McNeil, D. R. (1968). A solution to the fixed-cycle traffic light problem for compound poisson arrivals. *Journal of Applied Probability*, 5(3):624--635.

Mershad, K., Artail, H., and Gerla, M. (2012). Roamer: Roadside units as message routers in vanets. *Ad Hoc Networks*, 10(3):479 – 496. ISSN 1570-8705.

Meyn, S. P. and Tweedie, R. L. (2009). *Markov chains and stochastic stability*. Cambridge University Press.

Mishra, T., Garg, D., and Gore, M. (2011). A publish/subscribe communication infrastructure for vanet applications. In *Advanced Information Networking and Applications (WAINA), 2011 IEEE Workshops of International Conference on*, pages 442–446.

Montana, D. J. and Czerwinski, S. (1996). Evolving control laws for a network of traffic signals. In *Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation*, GECCO '96, pages 333--338, Cambridge, MA, USA. MIT Press.

Myerson, R. B. (2013). *Game theory*. Harvard university press.

Nagatani, T. and Hino, Y. (2014). Driving behavior and control in traffic system with two kinds of signals. *Physica A: Statistical Mechanics and its Applications*, 403(0):110 – 119. ISSN 0378-4371.

Nakatsuyama, M., Nagahashi, H., and Nishizuka, N. (1984). Fuzzy logic phase controller for traffic junctions in the one-way arterial road. In *Proceedings of the IFAC Ninth Triennial World Congress*, pages 2865--2870. Pergamon Press, Oxford.

Nekoui, M., Eslami, A., and Pishro-Nik, H. (2008). The capacity of vehicular ad hoc networks with infrastructure. In *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks and Workshops, 2008. WiOPT 2008. 6th International Symposium on*, pages 267–272.

Niittymäki, J. and Pursula, M. (2000). Signal control using fuzzy logic. *Fuzzy Sets and Systems*, 116(1):11 – 22. ISSN 0165-0114.

Oliveira, D. and Bazzan, A. (2006). Traffic lights control with adaptive group formation based on swarm intelligence. In Dorigo, M., Gambardella, L., Birattari, M., Martinoli, A., Poli, R., and Stutzle, T., editors, *Ant Colony Optimization and Swarm Intelligence*, volume 4150 of *Lecture Notes in Computer Science*, pages 520–521. Springer Berlin Heidelberg.

Oliveira, S., Teixeira, F., Macedo, D., Aquino, A. L. L., Lima, D. H. S., Silva, C., Silva, R. I., and Shiroma, P. M. (2013). Sistema de coleta e disseminação de dados de trânsito. In *Salão de Ferramentas. Simpósio Brasileiro de Redes de Computadores*. SBRC 2013.

Ormont, J., Walker, J., Banerjee, S., Sridharan, A., Seshadri, M., and Machiraju, S. (2008). A city-wide vehicular infrastructure for wide-area wireless experimentation. In *Proceedings of the Third ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization*, WiNTECH '08, pages 3--10, New York, NY, USA. ACM.

Osborne, M. J. and Rubinstein, A. (1994). *A course in game theory*. MIT press.

Ott, J. and Kutscher, D. (2004). Drive-thru internet: Ieee 802.11 b for" automobile" users. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 1. IEEE.

Palazzi, C. E., Ferretti, S., and Roccetti, M. (2009). Smart access points on the road for online gaming in vehicular networks. *Entertainment Computing*, 1(1):17 – 26. ISSN 1875-9521.

Palazzi, C. E., Roccetti, M., Ferretti, S., and Frizzoli, S. (2008). How to let gamers play in infrastructure-based vehicular networks. In *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology*, ACE '08, pages 95--98, New York, NY, USA. ACM.

Pappis, C. P. and Mamdani, E. H. (1977). A fuzzy logic controller for a traffic junction. *Systems, Man and Cybernetics, IEEE Transactions on*, 7(10):707–717. ISSN 0018-9472.

Patil, P. and Gokhale, A. (2013). Voronoi-based placement of road-side units to improve dynamic resource management in vehicular ad hoc networks. In *Collaboration Technologies and Systems (CTS), 2013 International Conference on*, pages 389–396.

Plobl, K. and Federrath, H. (2008). A privacy aware and efficient security infrastructure for vehicular ad hoc networks. *Computer Standards & Interfaces*, 30(6):390 – 397. ISSN 0920-5489. Special Issue: State of standards in the information systems security area.

Policy, E. T. (2001). White paper, european transportation policy for 2010:time to decide. In *Office for Official Publications of the European Communities, Luxembourg, Tech. Rep.*

Ramani, I. and Savage, S. (2005). Syncscan: practical fast handoff for 802.11 infrastructure networks. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 1, pages 675–684 vol. 1. ISSN 0743-166X.

Reis, A., Sargento, S., and Tonguz, O. (2011). On the performance of sparse vehicular networks with road side units. In *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, pages 1–5. ISSN 1550-2252.

Rosi, U., Hyder, C., and hoon Kim, T. (2008). A novel approach for infrastructure deployment for vanet. In *Future Generation Communication and Networking, 2008. FGCN '08. Second International Conference on*, volume 1, pages 234–238.

Rybick, J., Scheuermann, B., Kiess, W., Lochert, C., Fallahi, P., and Mauve, M. (2007). Challenge: peers on wheels - a road to new traffic information systems. In *Proc. 13th annual ACM international conference on Mobile computing and networking (MobiCom 2007)*, pages 215–221.

Shihuang, X. D. F. J. S. (1992). A fuzzy controller of traffic systems and its neural network implementation [j]. *Information and Control*, 2:001.

Shumao, O., Kun, Y., Hsiao-Hwa, C., and Alex, G. (2009). A selective downlink scheduling algorithm to enhance quality of vod services for wave networks. *EURASIP Journal on Wireless Communications and Networking*, 2009.

Silva, C., Oliveira, S., Aquino, A., and Teixeira, F. A. (2013). Pmcp: Uma heurística probabilística para otimizar a instalação de pontos de disseminação em redes veiculares. In *SBCUP 2013*.

Silva, C. M., Aquino, A. L., and Jr., W. M. (2015a). Deployment of roadside units based on partial mobility information. *Computer Communications*, (0):–. ISSN 0140-3664.

Silva, C. M., Aquino, A. L., and Meira, W. (2014a). Design of roadside infrastructure for information dissemination in vehicular networks. In *Network Operations and Management Symposium (NOMS), 2014 IEEE*, pages 1–8.

Silva, C. M., Aquino, A. L., and Meira, W. (2015b). Planning the roadside infrastructure for a small town with sparse traffic. In *ISCC, 2015 IEEE*, pages 1–8.

Silva, C. M., Aquino, A. L., and Meira Jr, W. (2014b). Probabilistic deployment of dissemination points in urban areas to support vehicular communication. *Journal of Applied Computing Research*, 3(1):34--41.

Silva, C. M., Aquino, A. L., and Meira Jr, W. (2015c). Smart traffic light for low traffic conditions: A solution for improving the drivers safety. *ACM/Springer Mobile Networks & Applications - Special Issue on Advances on Vehicular Communication Systems (accepted in 01/19/2015)*, -(-):–.

Silva, C. M. and Meira Jr, W. (2015a). Hybrid deployment: Integrating stationary and mobile roadside units. *Ad Hoc Networks (to appear)*, -(-):–.

Silva, C. M. and Meira Jr, W. (2015b). Roadside units deployment assisted by the full projection of the flow: A non intrusive strategy for planning the infrastructure for vehicular networks. *ACM/Springer Mobile Networks & Applications - Network Protocols and Algorithms for Vehicular Ad Hoc Networks (to appear)*, -(-):–.

Simonnard, M. and Jewell, W. S. (1966). Linear programming. In *Linear programming*. Prentice-Hall.

Smaldone, S., Han, L., Shankar, P., and Iftode, L. (2008). Roadspeak: enabling voice chat on roadways using vehicular social networks. In *Proceedings of the 1st Workshop on Social Network Systems*, pages 43--48. ACM.

Sommer, C., Eckhoff, D., and Dressler, F. (2013). Ivc in cities: Signal attenuation by buildings and how parked cars can improve the situation. *Mobile Computing, IEEE Transactions on*, PP(99):1–1. ISSN 1536-1233.

Sou, S.-I. (2010). A power-saving model for roadside unit deployment in vehicular networks. *Comm. Letters.*, 14(7):623--625. ISSN 1089-7798.

Sou, S.-I. and Tonguz, O. (2011). Enhancing vanet connectivity through roadside units on highways. *Vehicular Technology, IEEE Transactions on*, 60(8):3586–3602. ISSN 0018-9545.

Souza, F., Silva, C. M., and Guimaraes, C. (2009). Security in wireless networks. *e-Xacta Scientific Initiation Magazine*, 2.

Taliwal, V., Jiang, D., Mangold, H., Chen, C., and Sengupta, R. (2004). Empirical determination of channel characteristics for dsrc vehicle-to-vehicle communication. In *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*, pages 88--88. ACM.

Tayal, S. and Triphathi, M. R. (2012). Vanet-challenges in selection of vehicular mobility model. In *Proceedings of the 2012 Second International Conference on Advanced Computing & Communication Technologies*, ACCT '12, pages 231--235, Washington, DC, USA. IEEE Computer Society.

Teixeira, F., Silva, V., Leoni, J., Macedo, D., and Nogueira, J. M. S. (2014). Vehicular networks using the {IEEE} 802.11p standard: An experimental analysis. *Vehicular Communications*, 1(2):91 – 96. ISSN 2214-2096.

Teng, S.-H. (1995). Mutually repellant sampling. In Du, D.-Z. and Pardalos, P., editors, *Minimax and Applications*, volume 4 of *Nonconvex Optimization and Its Applications*, pages 129–140. Springer US.

Theraulaz, G., Bonabeau, E., and Denuebourg, J.-N. (1998). Response threshold reinforcements and division of labour in insect societies. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 265(1393):327–332.

Thompson, C., White, J., Dougherty, B., Albright, A., and Schmidt, D. C. (2010). Using smartphones to detect car accidents and provide situational awareness to emergency responders. In *Mobile Wireless Middleware, Operating Systems, and Applications*, pages 29--42. Springer.

Tonguz, O. (2011). Notice of violation of ieee publication principles biologically inspired solutions to fundamental transportation problems. *Communications Magazine, IEEE*, 49(11):106–115. ISSN 0163-6804.

Tonguz, O. and Viriyasitavat, W. (2013). Cars as roadside units: a self-organizing network solution. *Communications Magazine, IEEE*, 51(12):112–120. ISSN 0163-6804.

Tonguz, O., Viriyasitavat, W., and Bai, F. (2009). Modeling urban traffic: A cellular automata approach. *Communications Magazine, IEEE*, 47(5):142–150. ISSN 0163-6804.

Trullols, O., Fiore, M., Casetti, C., Chiasserini, C., and Ordinas, J. B. (2010). Planning roadside infrastructure for information dissemination in intelligent transportation systems. *Computer Communications*, 33(4):432 – 442. ISSN 0140-3664.

Trullols-Cruces, O., Fiore, M., and Barcelo-Ordinas, J. (2012). Cooperative download in vehicular environments. *Mobile Computing, IEEE Transactions on*, 11(4):663--678.

Vinel, A. (2012). 3gpp lte versus ieee 802.11p/wave: Which technology is able to support cooperative vehicular safety applications? *Wireless Communications Letters, IEEE*, 1(2):125–128. ISSN 2162-2337.

Viriyasitavat, W., Bai, F., and Tonguz, O. (2010). Uv-cast: An urban vehicular broadcast protocol. In *Vehicular Networking Conference (VNC), 2010 IEEE*, pages 25–32. ISSN 2157-9857.

Weijermars, W. A. M. (2007). *Analysis of urban traffic patterns using clustering.* University of Twente.

Wisitpongphan, N., Bai, F., Mudalige, P., and Tonguz, O. (2007). On the routing problem in disconnected vehicular ad-hoc networks. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pages 2291–2295. ISSN 0743-166X.

Wu, H., Fujimoto, R., Hunter, M., and Guensler, R. (2005a). An architecture study of infrastructure-based vehicular networks. In *Proceedings of the 8th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, pages 36--39. ACM.

Wu, H., Palekar, M., Fujimoto, R., Lee, J., Ko, J., Guensler, R., and Hunter, M. (2005b). Vehicular networks in urban transportation systems. In *Proceedings of the 2005 National Conference on Digital Government Research*, dg.o '05, pages 9--10. Digital Government Society of North America.

Wu, T.-J., Liao, W., and Chang, C.-J. (2012a). A cost-effective strategy for road-side unit placement in vehicular networks. *Communications, IEEE Transactions on*, 60(8):2295–2303. ISSN 0090-6778.

Wu, Y., Zhu, Y., and Li, B. (2012b). Infrastructure-assisted routing in vehicular networks. In *INFOCOM, 2012 Proceedings IEEE*, pages 1485--1493. IEEE.

Xie, B., Xia, G., Chen, Y., and Xu, M. (2013). Roadside infrastructure placement for information dissemination in urban its based on a probabilistic model. In *Network and Parallel Computing*, volume 8147 of *Lecture Notes in Computer Science*, pages 322–331. Springer Berlin Heidelberg.

Xiong, Y., Ma, J., Wang, W., and Tu, D. (2013). Roadgate: Mobility-centric roadside units deployment for vehicular networks. *International Journal of Distributed Sensor Networks. Hindawi Publishing Corporation. Article ID 690974.*, 2013.

Yan, T., Zhang, W., Wang, G., and Zhang, Y. (2014). Access points planning in urban area for data dissemination to drivers. *Vehicular Technology, IEEE Transactions on*, 63(1):390–402. ISSN 0018-9545.

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

Yu, Z., Teng, J., Bai, X., Xuan, D., and Jia, W. (2011). Connected coverage in wireless networks with directional antennas. In *INFOCOM, 2011 Proceedings IEEE*, pages 2264–2272. ISSN 0743-166X.

Zaki, M. J. and Meira Jr, W. (2014). *Data Mining and Analysis: Fundamental Concepts and Algorithms*. Cambridge University Press.

Zaldivar, J., Calafate, C. T., Cano, J. C., and Manzoni, P. (2011). Providing accident detection in vehicular networks through obd-ii devices and android-based smartphones. In *Local Computer Networks (LCN), 2011 IEEE 36th Conference on*, pages 813--819. IEEE.

Zhang, W., Chen, Y., Yang, Y., Wang, X., Zhang, Y., Hong, X., and Mao, G. (2012). Multi-hop connectivity probability in infrastructure-based vehicular networks. *Selected Areas in Communications, IEEE Journal on*, 30(4):740–747. ISSN 0733-8716.

Zhang, Y., Zhao, J., and Cao, G. (2007). On scheduling vehicle-roadside data access. In *Proceedings of the Fourth ACM International Workshop on Vehicular Ad Hoc Networks*, VANET '07, pages 9--18, New York, NY, USA. ACM.

Zhang, Y., Zhao, J., and Cao, G. (2010). Service scheduling of vehicle-roadside data access. *Mob. Netw. Appl.*, 15(1):83--96. ISSN 1383-469X.

Zheng, Z., Lu, Z., Sinha, P., and Kumar, S. (2010). Maximizing the contact opportunity for vehicular internet access. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. ISSN 0743-166X.

Zheng, Z., Sinha, P., and Kumar, S. (2009). Alpha coverage: Bounding the interconnection gap for vehicular internet access. In *INFOCOM 2009, IEEE*, pages 2831–2835. ISSN 0743-166X.

Appendix A

Non-Stop Driving-Thru Low Traffic Intersections

Projection of flow allows us to identify regions on the road network containing higher expected number of vehicles, which raises the opportunity to apply our model for supporting several kinds of vehicular and Intelligent Transportation Systems applications. When considering urban scenarios, traffic lights are highlighted as one of the most important devices for mobility: (a) traffic lights directly manage the flow of vehicles; (b) they are deployed in privileged locations; (c) they counting on energy supply; (d) they are visible to drivers; and, (e) several of them are already served with wired high speed networks.

The research on Smart Traffic Lights dates back to 1970s. Several promising technologies and strategies have been evaluated since then, but all of them share a focus on high traffic conditions. Invariably, all proposed solutions aim to reduce the occurrence of congestions. Obviously, solutions to attenuate congestions are important for mobility. But, a truly Intelligent Transportation System cannot be restricted to manage only very high traffic conditions. Urban mobility is much more complex than that: an Intelligent Transportation System obligatory has to provide solutions for medium traffic and low traffic conditions, which are also part of the urban mobility, and very distinct scenarios from the chaotic congestions of rush hours.

Here we present a novel application to handle **low traffic intersections**. We have searched for similar solutions in scientific literature and banks of patents, and we just have not found any similar solution. Our solution (LaNPro) avoids the stop of vehicles at intersections in low traffic conditions. LaNPro is a security solution to preserve the physical integrity of drivers in countries with high social discrepancy. Server-side of the solution is deployed as a module of a smart traffic light and it senses

the presence of vehicles along the road through input devices (radars, cameras, road sensors, wireless communication) to assign the right of way. While any smart traffic light is able to manage low traffic intersections, we argue that they are not specialized devices to perform such task, and thus they may lack important optimizations. Main differential of our approach is management of low traffic conditions, and that involves several challenges. Results show that our proposal can ensure the non-stop crossing of intersections having an expected traffic volume equal or less than λ=0.1 vehicles per second, assuming intersections of 2, 3 and 4 lanes, road segments 200m long, intersections 10m wide, vehicles 5m long traveling in an average speed of μ=40km/h with standard deviation σ=4km/h.

A.1 Related Work

Traffic lights can be classified into **static** or **dynamic** (smart). Main characteristic of **static** traffic lights is pre-programming, which means that the duration of each phase is defined in advance (Liu, 2007). In order to improve operation, static traffic lights may also employ several static programs scheduled along the day. However, since the control of traffic is non-linear and non-deterministic, static methods may not achieve high performance. This has motivated the development of dynamic traffic lights that continuously adapt to the traffic demands using several strategies, namely:

Fuzzy logic uses a set of rules to determine the preferred action of a traffic signal based on several inputs. Typically, this strategy suffers of scalability problems. For instance, Karakuzu and Demirci (2010) propose a fuzzy-logic-based traffic controller. Controller selects 3 kinds of time intervals of a day (morning, noon, evening). In the experimental studies performed by the authors, the waiting time of vehicles, the number of vehicles that passed through in unit time, the response of the traffic light controller with respect to the length of queue have been investigated. Literature also shows several relevant works like (Niittymäki and Pursula, 2000; Nakatsuyama et al., 1984; Pappis and Mamdani, 1977; Chiu and Chand, 1993; Shihuang, 1992).

Swarm intelligence is an area of artificial intelligence which has been applied to many optimization problems, including traffic signal control (McKenney and White, 2013). For instance, Oliveira and Bazzan (2006) apply the swarm principle of task allocation to select signal plans based on a theoretical pheromone stimulus emitted by waiting vehicles. García-Nieto et al. (2012) use swarm particle optimization to search for optimal signal plans within a traffic network. Our literature investigation also has captured important efforts in (Bonabeau et al., 1999; Theraulaz et al., 1998).

Evolutionary Computation typically aims to find a solution for the entire network. However, this solution requires an extremely high amount of computing power and may face scalability problems. Efforts applying evolutionary computation can also be found in (Foy et al., 1992; Montana and Czerwinski, 1996). **Reinforced learning** has also been applied to traffic signal management (Kaelbling et al., 1996).

Now we highlight some recent academic efforts: McKenney and White (2013) propose a technique to control traffic signals that rely on traffic observations made by available sensor devices and local communication between traffic lights. Authors evaluate the solution using a 9×7 block section of downtown Ottawa, with over 50 signalized intersections requiring control. Results demonstrate that the solution increases average vehicle speed by 6.58%.

Nagatani and Hino (2014) notice that real traffic is typically controlled by either synchronized or delayed traffic lights (green wave). Authors propose to evaluate the results achieved when we control traffic using both strategies mixed within the urban area. This work is motivated by the observation that metal is improved when the mixture receives another type of atom in order to improve a given property. They present the nonlinear dynamic model for the vehicular motion and investigate the dynamical behavior of vehicular traffic in order to derive the fundamental diagram for the traffic flow using just 2 kinds of signals to control the intersections. Authors conclude that this combination is important and necessary in urban scenarios.

Hamza-Lup et al. (2008) propose techniques for augmenting ITS to improve and support homeland security through evacuation algorithms. We also highlight industry efforts: BMW Green Wave Project[1] demonstrates that smart traffic lights reduce up to 20% fuel consumption. Audi Travolution[2] proposes the use of smart traffic lights combined with vehicular communication to implement speed advisory that minimizes stops at traffic lights. Results show the economy of 0.02 liters of fuel per traffic light.

Our proposal differs from all the previous ones in the sense that we focus on controlling intersection under low traffic conditions.

A.2 Overview of LaNPro

LaNPro implements an algorithm that routes vehicles across low traffic intersections, managing these vehicles to safely cross the intersection without stopping. Because we are focusing on a very restricted scenario, LaNPro must be combined with a standard module that will control the lights when the intersection is operating under medium

[1] http://www.themotorreport.com.au/46137/
[2] http://www.audi.com/.

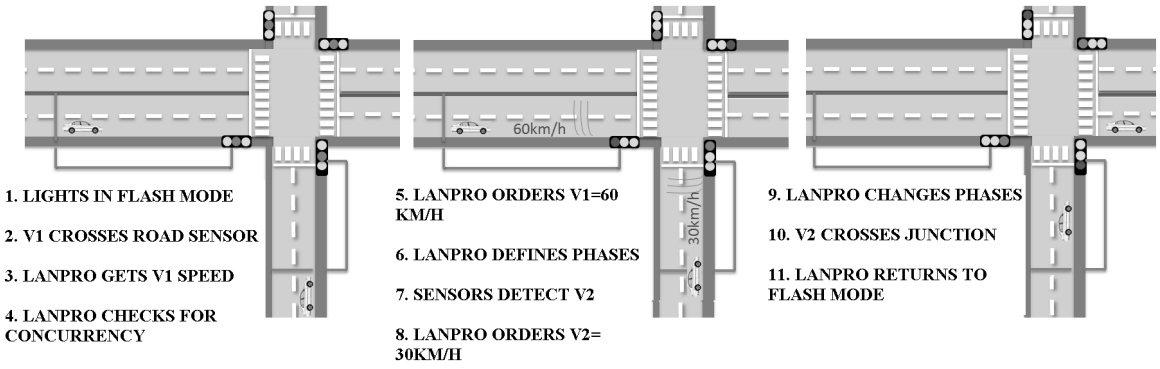


Figure A.2. Example of LaNPro operation.

or high traffic. Thus, the traffic light needs a scheduler that selects the active module: standard module or LaNPro (Figure A.2).

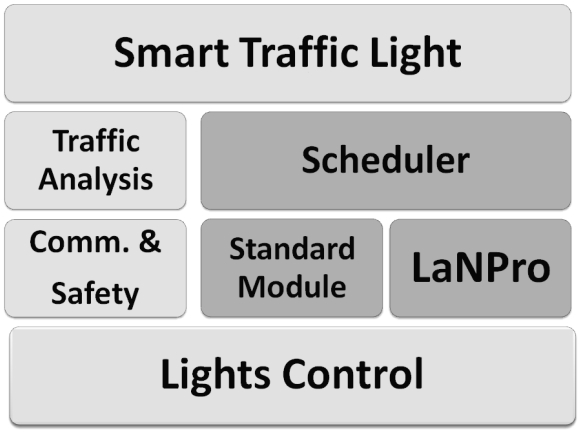


Figure A.1. Traffic light: scheduler selects between standard module and LaNPro according to the level of traffic.

Traffic light initializes under the control of standard module. When the traffic light detects low traffic (idleness) at the intersection, LaNPro is scheduled. LaNPro turns lights to blinking yellow (flash mode). Let's suppose an intersection of roads r_1 and r_2. LaNPro starts monitoring both segments of roads (r_1 and r_2) through a set of technologies (road sensors, cameras, radars) and any other possibly available (in fact, we envision LaNPro almost as an interface).

Figure A.2 presents the basic operation of LaNPro. When it detects a vehicle v_1 traveling on road r_1 heading to the intersection, LaNPro communicates to its pairs responsible for managing the intersection and requests priority for vehicle v_1. Because v_1 is the first vehicle detected, all traffic lights agree, v_1 receives green light and road

r_2 gets red light. As soon as vehicle v_1 traverses the intersection, LaNPro returns to blinking yellow if no more vehicles are heading to the intersection. However, if LaNPro detects vehicle v_2 on road r_2, LaNPro schedules v_2 to cross the intersection just after v_1 (v_1 was discovered earlier). Because LaNPro is monitoring v_1, LaNPro knows when v_1 will leave the intersection, and LaNPro computes the optimum speed to v_2. Vehicle v_2 receives the speed advisory and the expected time (in seconds) to receive green light. Driver adjusts the vehicle speed in order to cross the intersection without stopping. During all the operation LaNPro keeps monitoring the collision risk of vehicles. If there's a collision risk, LaNPro simply turns all phases to red in order to avoid a possible crash, and returns the control to the standard module. Similarly, if LaNPro detects a high volume of vehicles, it also returns the control to standard module. Simplified version of the scheduler is presented in Algorithm 10. It receives the following parameters:

λ: threshold of low traffic;

μ: interval (in minutes) for detecting low traffic;

η: maximum size of the vehicle queue.

Algorithm 10 Scheduler Algorithm.

Input: λ,μ, η;

```
1: while (true) do
2:     traffic ← 0
3:     schedule (standard_module())
4:     sleep (μ) minutes
5:     if (traffic ≤ λ) then
6:         schedule_and_wait (LaNPro(λ, η))
7:     end if
8: end while
```

Initially the scheduler selects standard module for operation during μ minutes. If after μ minutes the scheduler detects low traffic, it runs LaNPro. Whenever LaNPro detects any abnormal situation, it returns and the scheduler restarts standard module. In order to make this presentation as short as possible we are not considering important parameters to both the scheduler and LaNPro, like: (a) width of segments; (b) speed of vehicles; (c) distance of road sensors; (d) threshold of safety (discussed ahead); and others. LaNPro follows a different approach from typical smart traffic lights. Every time we think about improving the traffic, we think about rush hours and traffic jams. Main differential of our work is to improve low demand traffic, and this involves several

challenges. To the best of our knowledge, LaNPro is the first solution optimized for low traffic.

A.3 Main Functionalities of LaNPro

Organization of LaNPro is shown in Figure A.3. LaNPro has interface 'detection' responsible for receiving data of arriving vehicles. When any vehicle is detected, 'monitoring' module is activated to follow the vehicle until it traverses and leaves the intersection. Module 'events queue' manages the pending requests of intersection traversals issued by LaNPro: the allocation of the intersection is performed always by the master controller. Master controller is elected during startup. 'Priorization' module is intended to handle emergency vehicles and public transportation (cabs, BRT-bus rapid transportation). 'Alarm' module is responsible for identifying collision risks and to report risks to pairs: if a risk is identified, all traffic lights communicate between each other and start red light to all directions. Then, standard module is allocated to manage the intersection.

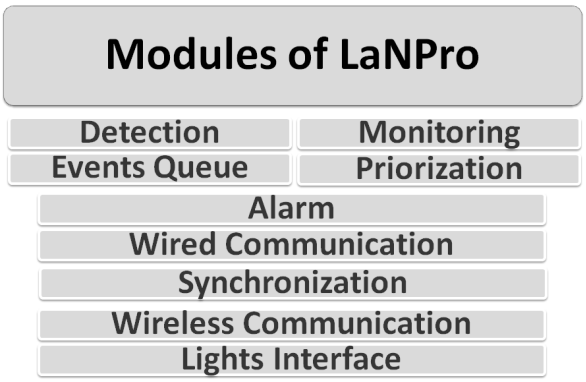


Figure A.3. LaNPro Organization.

'Wired communication' interacts with traffic light pairs. 'Wireless communication' is designed for vehicular communication. 'Synchronization' module is responsible for guaranteeing all pairs are working together, and the 'lights interface' receives commands from the master controller indicating actual phase.

A.4 Evaluation of the Collision Risk

Evaluation of the collision risk assuming that only 1 vehicle is traveling on each road segment in a intersection composed of 2 roads is trivial. We must make sure that

both vehicles will not reach the intersection at the same time, considering a safety threshold. Safety threshold indicates the difference in time (or distance) between 2 consecutive passages of competing cars through the intersection. This threshold is important because it mitigates small speed variations suffered by vehicles along the displacement and measurement errors of speed and position of each vehicle. Besides that, remember that drivers have to deal with unpredictable events. Thus, we cannot assume that speed advisory will always be followed.

A.5 Managing Intersections

Although LaNPro deals with situations of low traffic, the assumption that only 1 vehicle is present at each road segment is unreasonable. Our team has investigated 2 alternatives for managing low traffic intersections, namely: (a) slow switching; (b) fast switching.

(a) Slow switching: the traffic light tries to traverse as much vehicles as it can, until the first competitor vehicle arrives at the intersection. Then it reverses the phases. Competitor vehicles are advised to travel at the minimum speed allowed on the road[3]. This strategy has the advantage of being closer to current traffic signal standards.

(b) Fast switching: the traffic light schedules vehicles individually to traverse the intersection through the attribution of time slots. Whenever a vehicle is detected, it is assigned a time slot considering expected time to reach the intersection. In this work we investigate fast switching because it is a more challenging strategy.

A.6 LaNPro Algorithm

In this section we present a simplified version of LaNPro. Although LaNPro receives several parameters as input, the following algorithm concentrates on the most important ones.

λ**:** threshold of low traffic;

η**:** maximum size of the vehicle queue;

ϑ**:** id of the traffic light;

Λ**:** set of traffic light pairs.

Algorithm 11 LaNPro Algorithm.

Input: λ, η, ϑ, Λ;
1: ϑ.install_phase (BLINKING_YELLOW)
2: Λ.elect_master_controller (ϑ) ▷ the traffic light phases are controlled by a single entity.
3: $\Gamma \leftarrow \emptyset$ ▷ set holds detected vehicles.
4: start thread ϑ.thr_monitor (Γ) ▷ detection and monitoring of vehicles.
5: **while** (level $\leq \lambda$) And (queue $< \eta$) **do**
6: **if** ($\gamma = \vartheta$.thr_monitor.vehicle_detected) **then**
7: $\Gamma \leftarrow \Gamma \cup \gamma$ ▷ add vehicle to set of vehicles.
8: τ=master.create_ticket(γ) ▷ requests a new ticket from master controller.
9: ϑ.report_speed_advisory (γ, τ.get_speed()) ▷ reports speed advisory to vehicle.
10: **end if**
11: ϑ.install_phase (master.get_phase(ϑ)) ▷ master controller indicates phase.
12: **if** (ϑ.thr_monitor.high_traffic_detected) Or (ϑ.thr_monitor.vehicle_ignoring_speed_advisory) **then**
13: ϑ.install_phase (RED) **return** Λ.broadcast(finalize_lanpro(ϑ)) ▷ LaNPro is not meant to handle high traffic nor operate under the risk of collisions.
14: **end if**
15: **if** (κ=ϑ.thr_monitor.vehicle_leaving_intersection) **then**
16: $\Gamma = \Gamma - \kappa$ ▷ vehicle leaving the intersection does not require monitoring.
17: **end if**
18: **if** (ϑ.received_finalization_request) **then**
19: ϑ.install_phase (RED) **return** ▷ received request from a pair to finalize LaNPro.
20: **end if**
21: **end while**

Now we explain Algorithm 11 in details.

Line 1-4 Once LaNPro is scheduled, it turns lights to blinking yellow and elects a master controller that will define the phases to be adopted at each instant of time. Master controller also works as a ticket server, managing the allocation of the slots to vehicles. LaNPro stores the list of known vehicles in Γ. LaNPro also starts a thread to monitor vehicles along the intersection;

Line 5 LaNPro loops while the level of traffic is below λ and the queue of vehicles is less than η;

Lines 6-10 If LaNPro detects a vehicle heading to the intersection, the set of monitored vehicles (Γ) receives 1 more element (γ) and the traffic light requests the master controller to create a ticket for the new vehicle. Ticket assigns a time slot when the vehicle will cross the intersection. Given that LaNpro has the distance from γ to the intersection, it is able to compute the speed advisory for γ. When the computed speed is lower than minimum speed LaNPro assumes high traffic and quits;

Line 11 LaNPro requests phase to master controller and update lights;

Lines 12-15 If LaNPro detects high traffic or any vehicle ignoring the speed advisory, it returns the operation to standard module;

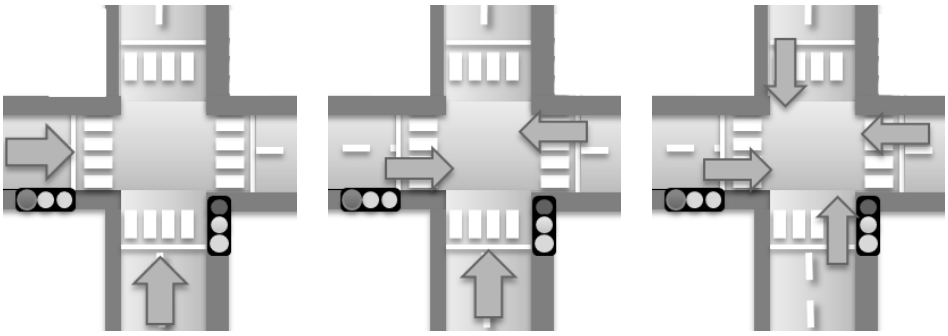
[3]Defined by the transit authority.

Lines 16-18 When any vehicle leaves the intersection, it is removed from the set of monitored vehicles (Γ);

Lines 19-22 If traffic light receives request to finalize operation from pairs[4], it installs red phase and returns execution to standard module.

A.7 Results

This section presents our simulations and results evaluating the potential contribution of LaNPro. Flows of vehicles are modeled via Poisson distribution supported by studies Chun et al. (2010); Larson (1982); McNeil (1968); Gerlough and Schuhl (1955). Poisson expresses the probability of a series of events occuring in a certain period of time given that such events are independent of when happened the last one. Vehicle speed is modeled on a Gaussian distribution Larson (1982) with parameters μ=40km/h and σ=4km/h. Segments of roads are 200m long, intersections are 10m wide. Results are based on 100 simulations of traffic, each one 5,000s long.



(a) 2 lanes. (b) 3 lanes. (c) 4 lanes.

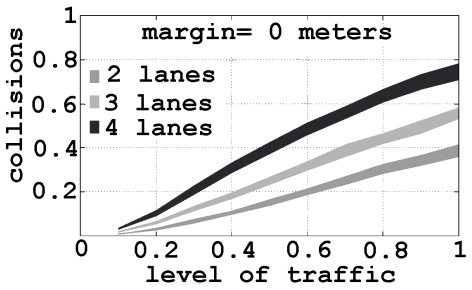
Figure A.4. Intersections considered during experiments: 2-4 lanes.

In order to evaluate LaNPro we built a simulator containing a single intersection consisting of 2, 3 and 4 lanes as indicated in Figure A.4. Simulation is composed of 3 categories of experiments:

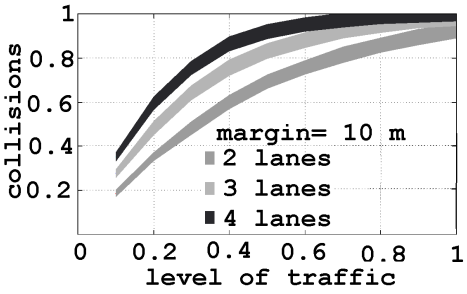
(i) intersection without traffic light: we measure number of collisions in blind traverse of vehicles with/without safety margins for different levels of traffic;

(ii) intersection signalized using traffic light with fixed plan: we measure number of vehicles stops in red lights for different levels of traffic;

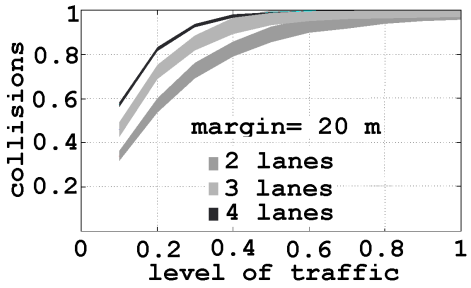
[4]'pairs' is a set of traffic lights responsible for an intersection.



(a) Collisions (no safety margin).



(b) Collisions (margin=10m).



(c) Collisions (margin=20m).

Figure A.5. Probability of collisions for intersections composed from 2 to 4 lanes as we change safety margin (0m; 10m; 20m).

(iii) intersection signalized using LaNPro: we measure number of LaNPro interruptions and queue size of vehicles, also for different levels of traffic.

A.7.0.1 Intersection without traffic light: collisions in blind traversal with no safety margin

In this experiment we evaluate the probability of collisions of vehicles traversing intersections in a non-stop and blind way. Basically, drivers maintain their route independently of any other variable. We present results for an intersection composed of 2 road segments, each road segment having 1 or 2 lanes as we vary traffic level. Figure A.5(a) indicates the probability of collision considering same level of traffic on the roads under study. The x-axis indicates level of traffic from $0 \leq \lambda \leq 1$, while y-axis indicates collision probability.

Curve 'A' indicates an intersection between 2 single lane roads. Curve 'B' indicates intersection between a single lane and a double lane road. Curve 'C' indicates intersection between 2 double lane roads. Width of each curve indicates the standard deviation. We consider as time unit the average time required for a vehicle to traverse the entire intersection. We can notice that when the traffic level is $\lambda = 1.0$ we achieve

A=40%, B=60% and C=80% of collisions. In practical terms, that means considering an intersection 10m wide, density of vehicles less than 1 vehicle per 10m:

- 40% of the vehicles collide in 2 single-lane roads;
- 60% collide if we consider 3 lanes;
- 80% collide in 4 lanes.

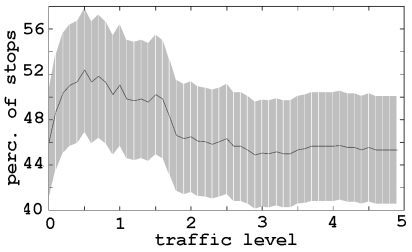
Notice that 1 vehicle per 10m is not exactly what we mean by low traffic: remember that a vehicle is approximately 5m long. Nevertheless, this result demonstrates that if we introduce some kind of ordering, we can manage the non-stop traversal of intersections efficiently[5]. However, cars are operated by humans and we cannot allow vehicles to cross over an intersection without a minimum distance (or delay) between consecutive passages. We call this minimum distance as **safety margin** or **safety threshold**. In the next experiment we measure the number of collisions considering distinct safety margins.

A.7.0.2 Intersection without traffic light: collisions in blind traverse with variable safety margin

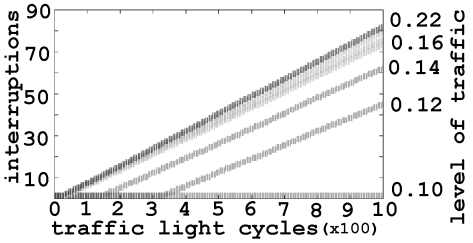
Basically, this is the same simulation as the previous one. Difference lies of the fact that now we add safety margins. If any vehicle enters the intersection in a distance less than some threshold, we consider the event as a collision. We arbitrarily define the safety margins as 10m and 20m. In Figure A.5 the x-axis represents the level of traffic from $0 \leq \lambda \leq 1$. The y-axis indicates the collisions probability. Plot presents the same combinations of intersections.

We notice that the volume of collisions increases very fast as we increase the level of traffic and/or the safety margin. As mentioned before, LaNPro manages traffic levels up to $\lambda = 0.10$ considering a safety margin of 20m. Contribution of LaNPro is to eliminate all stops when $\lambda \leq 0.10$. When we consider a safety margin of 20m, 58% of the vehicles will collide if they attempt to drive through the intersection without stopping. In the next simulation we signalize the intersection using a static traffic light and we compute the percentage of vehicles receiving red lights.

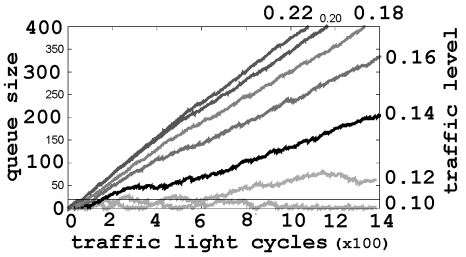
[5]In Section A.7.0.4 we demonstrate that LaNPro can manage the traversal of intersections up to $\lambda = 0.1$ vehicles per unit of time.



(a) Stops in static traffic lights.



(b) LaNPro interruptions.



(c) Queue size with LaNPro.

Figure A.6. (a) Evaluation of the amount of vehicles stopping on a red light when the intersection has a static traffic light: low traffic incurs in increased traffic light stops. (b) Interruption of LaNPro: LaNPro routes all traffic $\lambda \leq 0.10$. (c) Fluctuation of the queue of vehicles: LaNPro presents tiny queues for $\lambda = 0.10$, being able to route all traffic.

A.7.0.3 Intersection with fixed programming traffic light: percentage of vehicles stopping on red lights

In this experiment the intersection receives a traffic light with fixed programming (green time=1min; red time=1min) and we measure the percentage of vehicles that have to wait in red light. Both roads follow the same distribution of arrivals. Thus, we would expect 50% of stops. Figure A.6(a) shows the results and the standard deviation. The x-axis represents the traffic level $0 \leq \lambda \leq 5$. The y-axis represents the percentage of vehicles that need to wait in red light.

Symmetric timing shows very poor performance in low traffic situations. As the level of traffic increases, the percentage of stopped vehicles stabilizes around 45%. Standard deviation (σ) is indicated in the area surrounding the curve. Average stop time is μ=28s with σ=8s. An average stop time of 28s in phases of 1min represents too much idle time for drivers, and that is the reason why stopping at traffic lights is a situation of real danger. Such issue clarifies the motivation for LaNPro. In the next experiment we replace the static traffic light for LaNPro and we evaluate its performance.

A.7.0.4 LaNPro interruptions

Now we evaluate the number of times that LaNPro is interrupted due to: (a) high level of traffic; or (b) the impossibility to route the vehicles without stopping them at the traffic light. Every time a vehicle stops, LaNPro is interrupted and **standard module** is scheduled. LaNPro only returns when the traffic light detects idleness in the intersection. In order to reduce the simulation time, every time LaNPro is interrupted we flush the queue of vehicles and restart LaNPro. Our experiment considers Brazilian traffic rules: urban roads have a maximum speed of 40km/h and minimum of 20km/h.

Figure A.6(b) shows the result of the experiment. The x-axis indicates the amount of traffic light cycles, while the y-axis shows the number of times that LaNPro was interrupted. Arrival rate of vehicles is $0.10 \leq \lambda \leq 0.22$. Rates below 0.10 have not generated any LaNPro interruption. Thus, in this scenario LaNPro is able to avoid the stops of vehicles for arrival rates until $\lambda = 0.10$.

Let's make this result more practical: we assume intersection 10m long. Vehicles' speed is 40km/h (approximately 11m/s). Thus, the vehicle crosses the intersection in approximately 1s. The λ is computed considering the expected time period required for a vehicle to cross over the intersection. Considering all of this, the λ=0.10 means that LaNPro is able to route vehicles if they arrive according to the distribution of Poisson with λ equal to 1 vehicle per 10s considering a safety margin of 20m.

Routing 1 vehicle every 10s may seems easy at a first glance. However, the distribution of Poisson demonstrates that vehicles have a non-negligible probability of arriving in groups. In the last experiment we demonstrate how the queue of vehicles fluctuates when LaNPro is controlling the intersection.

A.7.0.5 Queue size of vehicles with LaNPro

In this experiment we evaluate the queue of vehicles when the intersection is controlled by LaNPro. A large queue inviabilizes the speed advisory mechanism for the absence of space. Thus, now we present an experiment that reveals the queue size of vehicles when using LaNPro. This experiment is also important to indicate LaNPro limitations. Figure A.6(c) presents a plot that demonstrates the queue of vehicles (sum of vehicles in all roads leading to the intersection) during the operation of LaNPro. The x-axis represents traffic light cycles (time), while the y-axis represents the queue size. Experiment is executed for arrival rates from $0.10 \leq \lambda \leq 0.22$. Plot indicates the queue size in each instant of time. Results demonstrate that arrival rates above λ=0.10 imply in queues, possibly leading to LaNPro interruptions.

A.8 Remarks

LaNPro is a module of a smart traffic light designed to operate under low traffic conditions usually found in small cities and during early hours of morning. Goal of LaNPro is to route the vehicles to cross over an intersection without stopping on red light. Our results demonstrate that when we consider a static traffic light controlling a low traffic intersection, 45% of vehicles receive red light. If we consider the duration of each phase to be 2min (red=1min; green=1min) the average stop time of drivers is 28s with standard deviation of 8s, a very high waiting time that favors criminal activities in traffic lights in countries with high social discrepancy.

Main motivation of LaNPro is not reducing travel time nor fuel consumption. **LaNPro is a solution for security** of drivers and passengers. We have conducted an extensive research in scientific literature and banks of patents, and as far as we are concerned, LaNPro is the first solution to address low traffic intersections. Technical challenge of LaNPro is to deal with variable instantaneous concentrations of vehicles that may happen, as ruled by the Poisson process.

In this section we have studied the viability of LaNPro and we conclude that it can handle an arrival rate of 1 vehicle per 10s supposing intersections 10m wide, road segments 200m long, junctions composed of 4 lanes and safety margin of 20m between each passage (slot size) considering **fast switching** mechanism (Section A.5). While LaNPro imposes some interesting engineering challenges, it also provides improvements in the urban mobility. Now we are investigating the capabilities of low switching. While research on smart traffic lights continues to address high traffic, we highlight that 'low traffic' is a natural part of urban dynamics, and it must be properly addressed as well. Real Inteligent Transportation Systems will only be possible if we learn how to manage both 'high traffic' and 'low traffic'.

Appendix B

Publications & Awards

B.1 Award

- Best-paper Brazilian Symposium of Ubiquitous and Pervasive Computing - SBCUP'2013.
- Best-paper Brazilian Symposium of Ubiquitous and Pervasive Computing - SBCUP'2014.

B.2 Patent Application

- **BR 10 2013 027283 3**: Dispositivo de controle de cruzamentos de trânsito com baixo fluxo de veículos automotores (Device for controlling low traffic intersections).

B.3 Journals

- Silva, C. M.; Aquino, A. L. L.; Meira, W.; **Probabilistic Deployment of Dissemination Points in Urban Areas to Support Vehicular Communication**; Journal of Applied Computing Research - JACR, 3(1):34-41; doi: 10.4013/jacr.2013.31.04; January-June 2013.
- Silva, C. M.; Aquino, A. L. L.; Meira, W.; **Smart Traffic Light for Low Traffic Conditions: A Solution for Improving Safety of Drivers**; ACM/Springer Mobile Networks & Applications - Special Issue on Advances on Vehicular Communication Systems.

- Silva, C. M.; Aquino, A. L. L.; Meira, W.; **Deployment of Roadside Units Based on Partial Mobility Information**; Computer Communications (Elsevier).

B.4 International Conferences

- Silva, C. M.; Aquino, A. L. L.; Meira, W.; **Design of Roadside Infrastructure for Information Dissemination in Vehicular Networks**; IEEE/IFIP Network Management and Operational Symposium - NOMS'2014, Krakow-Poland.

- Faraj, M. F.; Sarubbi, J. F.; Silva, C. M.; Porto, M. F.; Nunes, N. T.; **A Real Geographical Application for the School Bus Routing Problem**; 17th International IEEE Conference on Intelligent Transportation Systems - ITSC'2014, Qingdao, China.

B.5 Brazilian Conferences

- Silva, C. M.; Oliveira, S.; Aquino, A. L. L.; Teixeira, F.; **PMCP: Uma Heurística Probabilística para Otimizar a Instalação de Pontos de Disseminação em Redes Veiculares**; Simpósio Brasileiro de Computação Ubíqua - SBCUP'2013, Maceió-AL.

- Oliveira, S.; Teixeira, F.; Macedo, D.; Aquino, A. L. L.; Lima, D. H. S.; Silva, C. M.; Silva, R. I.; Shiroma, P.; **Sistema de Coleta e Disseminação de Dados de Trânsito**; Simpósio Brasileiro de Redes de Computadores - SBRC'2013, Brasília-DF.

- Silva, C. M.; Aquino, A. L. L.; Meira, W.; **Non-stop Drive-through Low Traffic Intersection**; Simpósio Brasileiro de Computação Ubíqua - SBCUP'2014, Brasília-DF.

- Faraj, M. F.; Sarubbi, J. F.; Silva, C. M.; Porto, M. F.; Nunes, N. T.; **Estudo de Caso: O Problema do Transporte Escolar Rural em Minas Gerais**; XLVI Brazilian Symposium of Operational Research - SBPO'2014.

Appendix C

Complementary Study of the Literature

We have collect data over 110 articles, and after the application of the exclusion criteria, 68 works were selected: in this step we have discarded articles not addressing aspects of infrastructure-based vehicular networks. Thus, our full sample consists of 68 articles from IEEE, ACM and Elsevier.

Figure C.1 presents the distribution of articles over years: number of articles addressing infrastructure-based vehicular networks increases over the years. Figure C.2 characterizes our sample in terms of publishers: 65% from IEEE; 23% from ACM; 12% from Elsevier. Figure C.3 indicates the distribution of articles per publisher and year: until 2007 ACM lead the publication of articles. After 2008 IEEE represents majority of published works.

C.1 Classification

Our sample contains articles dealing with 4 macro subjects:

- **Architecture**: works proposing new network organizations, testbeds, proofs of concept, fields experimentations and hardware studies;
- **Communication**: works with a clear focus on aspects of communication, like: protocols, data dissemination strategies, routing, connectivity, low level aspects of communication, channel allocation, handoff strategies, throughput of the network, quality of service, real-time messaging, studies of multi-hop dissemination;

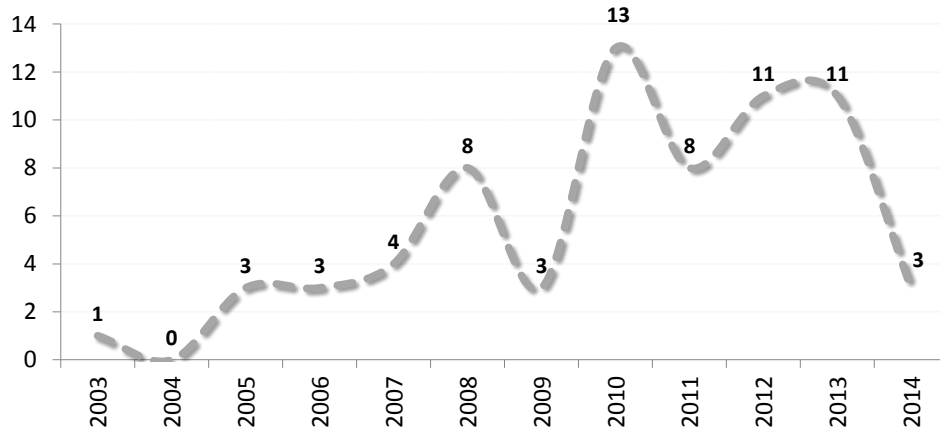


Figure C.1. Published Articles per Year (Jan. 2003 to Feb. 2014). The x-axis indicates year, while y-axis indicates number of published works composing our sample.

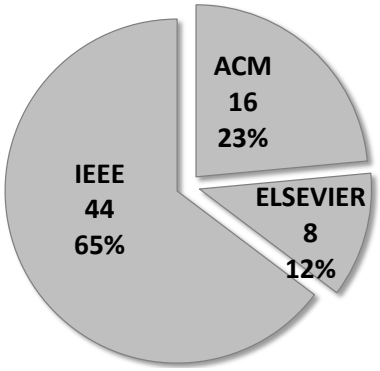


Figure C.2. Articles per Publisher (Jan. 2003 to Feb. 2014). Figure indicates number of works and percentage according to total number of works composing our sample.

- **Deployment**: works proposing strategies to physically locate the roadside units[1], theoretical studies discussing metrics and strategies to evaluate the deployment, and theoretical studies about requirements, features or properties of the deployment.

Because we describe 68 works in this document, the non-adoption of some classification scheme would undermine the understanding of this document. Nevertheless, because every work in our sample addresses some aspect of infrastructure for vehicular networks, it is expected that some works may belong to more than one category (class).

[1]Roadside units are communication devices deployed over the road network with the goal to support vehicular communication.

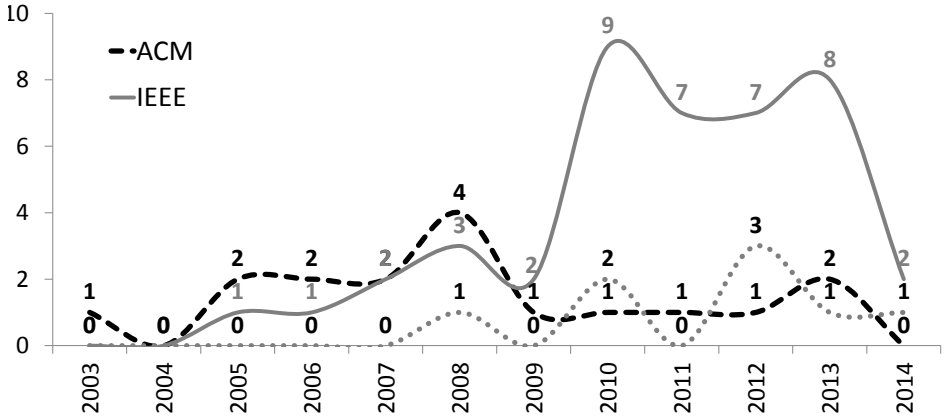


Figure C.3. Evolution of Articles per Publisher (Jan. 2003 to Feb. 2014). The x-axis indicates year, while y-axis indicates number of published articles. Figure shows the evolution for ACM, IEEE, and Elsevier.

For instance: Lochert et al. propose in (Lochert et al., 2008) both a data aggregation mechanism (class 'communication') and a deployment algorithm (class 'deployment'). When facing these situations, we evaluate the most important contribution of the research (in our particular opinion) to select the best class for the work. Particularly, we have considered the work of Lochert et al. as a deployment strategy powered by a data aggregation engine. Obviously, there may be some necessary and natural discussion about such classification. But, we believe we have reached a consistent set of classes for the purposes of this text. Figure C.4 indicates the number of works per class.

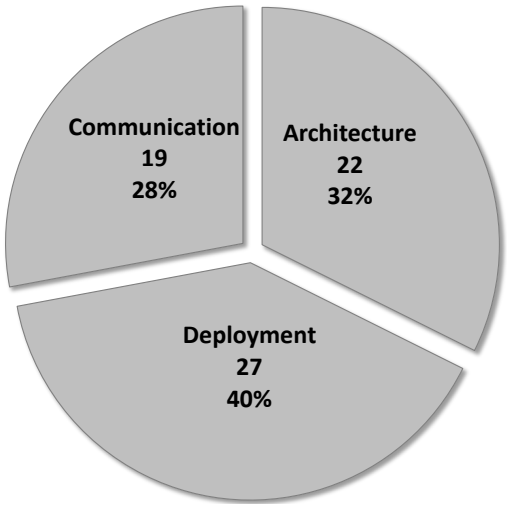


Figure C.4. Number of articles per class: 27 works proposing deployment strategies; 22 works investigating architectures; and, 19 works proposing solutions for improving communication.

Research community is aware that costs of the infrastructure are considerable. Thus, several research efforts are towards maximizing the coverage of roadside units through a more efficient 'deployment' (37% of the works) and new 'architectures' (35%) designed to maximize communication efficiency. Although wireless networks have been extensively studied in the past 20 years, high mobility imposed by vehicles demands complementary studies: articles addressing 'communication' represent 25% of total efforts. Figure C.5 indicates number of published works from Jan. 2003 to Feb. 2014: classes 'architecture' and 'communication' are stable in number of published works per year, while 'deployment' shows growing efforts.

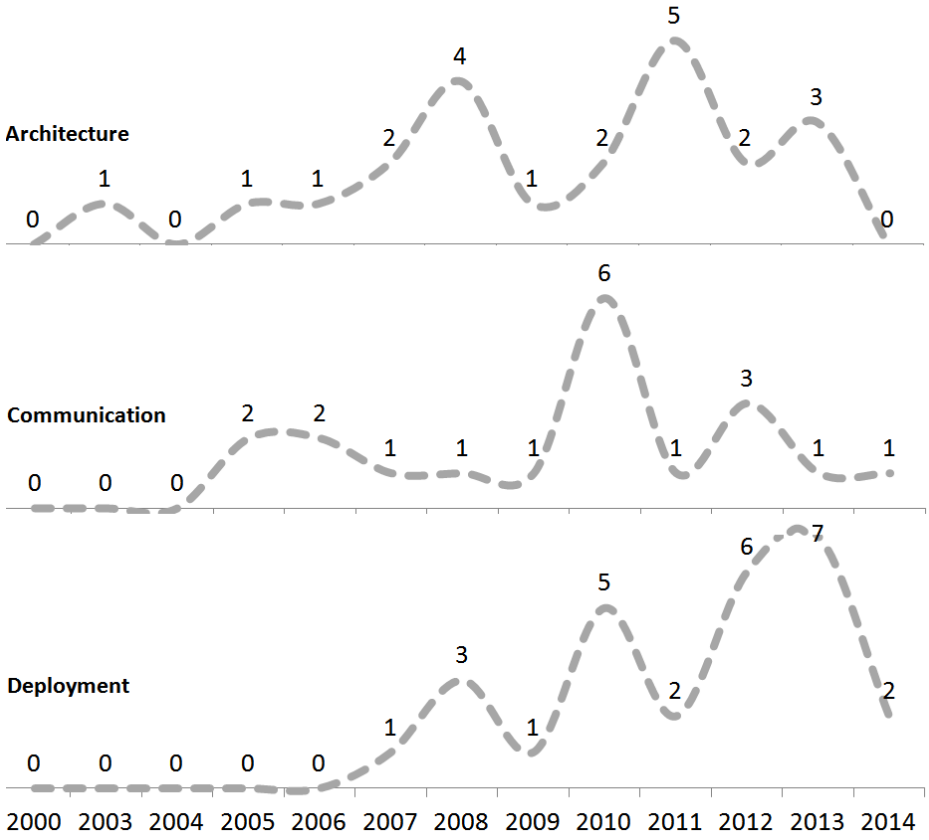


Figure C.5. Evolution of published articles per year (2003-2014): x-axis indicates year, while y-axis indicates number of works (architecture, deployment, and communication).

C.2 Overview of Research

In the early 2000s most of the vehicular research dealt with low level aspects of communication. High mobility of nodes and the absence of energy supply or power

restrictions brought several challenges to the community. The US Department of Transportation (USDOT) has invested millions of dollars to integrate vehicles and communication infrastructure in order to make the current transportation system more intelligent Zhang et al. (2007). As long as the community is solving low level problems, the research walks towards high level aspects of the vehicular communication. Frenkiel et al. Frenkiel et al. (2000) pioneered **Infostations**: vehicles engage in 'infofueling' as they opportunistically drive past roadside access points.

Research community is aware of the challenges that vehicular communication imposes, after all, communication channel must deal with several nodes (cars) attempting to send and receive data simultaneously, possibly at high speeds. ASTM[2] and IEEE[3] adopted DSRC[4] standard providing wireless communications capabilities for transportation applications within a 1000m range at typical highway speeds. DSRC provides 7 channels in the spectrum between 5.850 to 5.925 GHz, namely ITS-RS[5] licensed for ITS[6] applications, with different channels designated for different applications, and one specifically reserved for vehicle-to-vehicle communications Yousefi et al. (2006).

In order to present an high level overview of the research conducted along the years, we use a set of *wordles* displayed before the description of the researches. Such *wordles* are built using the abstract section of the selected articles.

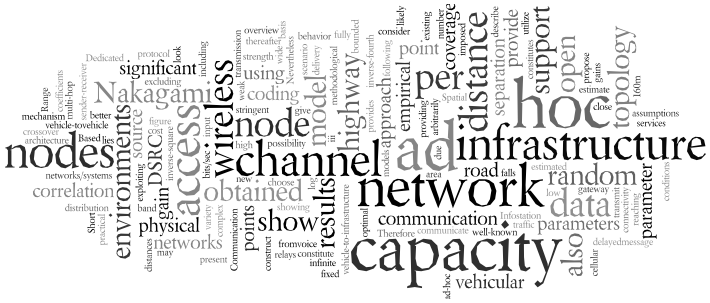


Figure C.6. Wordle of 2003-2004 abstracts.

In 2003-2004 (Figure C.6) researchers are focused in low level details (physical/MAC layers) of the vehicular communication: Kozat et al. Kozat and Tassiulas (2003) investigate the capacity of the wireless channel, while Taliwal et al. Taliwal et al. (2004) investigate channel access.

Most common terms: *channel, access, capacity, ad hoc, physical, nodes.*

[2]ASTM-American Society for Testing and Materials. http://www.astm.org/

[3]Institute of Electrical and Electronics Engineers.

[4]DSRC-Dedicated Short Range Communication. http://www.its.dot.gov/DSRC/

[5]Intelligent Transportation Systems Radio Service.

[6]Intelligent Transportation Systems.

Figure C.7. Wordle of 2005-2006 abstracts.

In 2005-2006 (Figure C.7) most of the research is still on evaluating low level architectures of communication. In 2006 the dominant topic are aspects of the ad hoc vehicle-to-vehicle communication. But the research community visualizes the need to offer Internet access to vehicles. Thus, researchers investigate architecture proposals in Wu et al. (2005a); Gerla et al. (2006), handoff strategies in Ramani and Savage (2005); Brik et al. (2005); Amir et al. (2006), network throughput in Hadaller et al. (2006) and multi-hop communication in Korkmaz et al. (2006).

Most common terms: *protocol, throughput, wireless, access, internet, vehicle.*

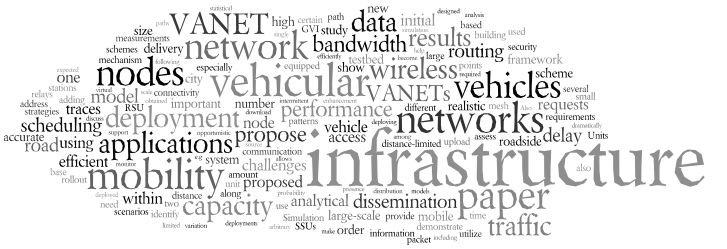


Figure C.8. Wordle of 2007-2008 abstracts.

In 2007-2008 (Figure C.8) research community turns attention to validity of the mobility models employed in vehicular simulations and strategies for vehicles' localization. They also start to exploit possibles uses of an infrastructure for vehicular networks in terms of security and applications. Community demonstrates interest in the potential use of public Wi-Fi access points to enhance vehicular communication, and we have initial works handling deployment of infrastructure. We can also find two surveys addressing VANETs: Hartenstein and Laberteaux Hartenstein and Laberteaux (2008) turn attention to vehicle-to-vehicle communication and the impact of decentralization, channel access, market issues, security, privacy and validation of simulations. Jakubiak and Koucheryavy Jakubiak and Koucheryavy (2008) discuss limits of vehicular communication. Both works are important as they picture the main

challenges of that time. In fact, some of these challenges remain unanswered until today, such as security, economic issues and channel access during rush. Infrastructure gains visibility as researchers realize its importance to support the dissemination of data in vehicular networks Lochert et al. (2007). For a matter of costs, some researches focus on alternative and less-expensive methods to achieve the benefits of a dedicated infrastructure. Works investigate publicly available Wi-Fi's Marfia et al. (2007) and adoption of virtual infrastructures Lochert et al. (2008); Jerbi et al. (2008). At the same time, Palazzi et al. Palazzi et al. (2008) propose a smart infrastructure to enable gamming and video streaming.

Community also realizes that bringing Internet access to drivers enables the development of a myriad of vehicular applications and traffic information systems to perform several tasks. Examples of these solutions are: (a) Pothole Patrol Eriksson et al. (2008) to monitor roads conditions; (b) Waze Mobile App; (c) Peer on Wheels Rybick et al. (2007) to monitor traffic conditions; and (d) RoadSpeak Smaldone et al. (2008) to enable chatting between drivers. Performance of the network is deeply studied in works Khabazian and Mehmet Ali (2007); Zhang et al. (2007); Wisitpongphan et al. (2007); Nekoui et al. (2008), and the available communication hardware is evaluated in order to discover optimal solutions Banerjee et al. (2008). Testbeds are proposed in Capone et al. (2007); Ormont et al. (2008) to evaluate practical issues of the vehicular communication, while requirements of privacy and security of the infrastructure are addressed in Plobl and Federrath (2008).

Fiore and Harri Fiore and Härri (2008) demonstrate that mobility has a great impact over simulation results. Boukerche et al. Boukerche et al. (2008) address localization of vehicles and suggests data fusion techniques will play an important role in the future, while Rosi et al. (2008) exploit the localization of vehicles using the infrastructure.

Most common terms: *infrastructure, vehicular, applications, data, vanet, mobility, performance, traffic, deployment.*

Figure C.9. Wordle of 2009-2010 abstracts.

In 2009-2010 (Figure C.9) 'rsu' (roadside unit) is highlighted in the *wordle*. Over 40% of articles are addressing infrastructure deployment. Efforts investigate deployment strategies using: heuristics Zheng et al. (2009); Trullols et al. (2010); Sou (2010); Lee and Kim (2010) and cluster-based approaches Kchiche and Kamoun (2010). Some works also propose evaluation metrics for deployment activities Zheng et al. (2010). A virtual infrastructure using buses is proposed in Luo et al. (2010), while a secure infrastructure is proposed in Fernandez Ruiz et al. (2010). When considering vehicle-to-infrastructure communication, we notice works addressing: data scheduling Shumao et al. (2009); Zhang et al. (2010), real-time communication Korkmaz et al. (2010); Liu and Lee (2010), delivery in sparse networks Abdrabou et al. (2010), analysis of connectivity probability Abdrabou and Zhuang (2009); Chun and Mao (2010); Chun et al. (2010), broadcast protocols Viriyasitavat et al. (2010), and a proposal for cooperative geo-routing Borsetti and Gozalvez (2010). Tonguz et al. Tonguz et al. (2009) address mobility by proposing a model to simulate urban traffic.
Most common terms: *network, infrastructure, performance, coverage, rsu.*

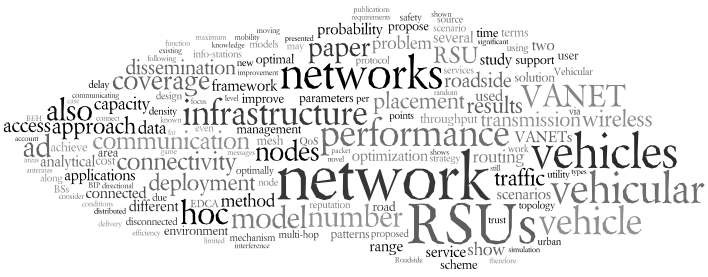


Figure C.10. Wordle of 2011-2012 abstracts.

In 2011-2012 (Figure C.10) articles addressing infrastructure deployment consider: probabilistic models Sou and Tonguz (2011); Abdrabou and Zhuang (2011); linear programming formulations Aslam et al. (2012); Liang et al. (2012); Wu et al. (2012a); heuristics Cataldi and Harri (2011); genetic algorithms Cavalcante et al. (2012), and Game theory Filippini et al. (2012). In terms of architecture, we notice analytic studies for selection of communication hardware Liang et al. (2012); Yu et al. (2011), light architectures Reis et al. (2011), virtual architectures Mishra et al. (2011), biologically inspired architectures Tonguz (2011). In terms of communication, we notice works addressing: infrastructure-based routing Annese et al. (2011); Mershad et al. (2012), cooperative data dissemination Liang and Zhuang (2012), QoS controlled media access Luan et al. (2012), multi-hop communication Zhang et al. (2012), content download Malandrino et al. (2011), and data traffic Mangel and Hartenstein (2011). Mobility is addressed in Koegel et al. (2011); Tayal and Triphathi (2012). The *wordle*

of 2011-2012 is much similar to that of 2009-2010, except for more visibility of 'rsu'.
Most common terms: *rsu, infrastructure, performance, model.*

Figure C.11. Wordle of 2013 - Feb.2014 abstracts.

In 2013 - Feb. 2014 (Figure C.11) we notice articles addressing high-level aspects of infrastructure-based vehicular networks: Tonguz and Viriyasitavat Tonguz and Viriyasitavat (2013) propose a self-organizing network using cars as roadside units. Luan et al. Luan et al. (2013) propose the use of roadside buffers, cheap devices for store-and-forward messages to passing vehicles. Sommer et al. Sommer et al. (2013) study signal attenuation by buildings and propose the use of parked cars to help the signal propagation. When we consider communication, Harigovindan et al. Harigovindan et al. (2014) develop a mechanism for fair channel allocation, while Bruno and Nurchis Bruno and Nurchis (2013) propose a mechanism to eliminate redundancy in data collected by vehicles in a distributed basis. Infrastructure deployment is addressed in several works: genetic approaches Cheng et al. (2013), voronoi diagrams Patil and Gokhale (2013), analytic studies Lu et al. (2013), randomized algorithms Liya et al. (2013), deployment optimized for content download Liu et al. (2013), bipartite graphs Yan et al. (2014), and intersection-priority Chi et al. (2013). Given that high mobility is a key feature of vehicular networks, Silva et al. Silva et al. (2014a) propose a deployment strategy using a model to describe the vehicular mobility.
Most common terms: *rsu, vehicles, problem, data, coverage, traffic, optimal, performance.*

C.3 Citations

Number of total citations per year is presented in Figure C.12. The x-axis indicates year, while the y-axis indicates the sum of citations from articles published that year. Peak of citations is achieved in 2006 (850 citations).

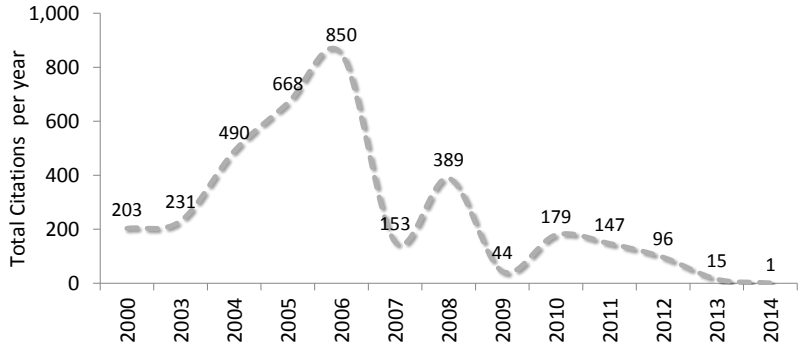


Figure C.12. Citations per year: x-axis indicates year, while y-axis indicates the sum of citations from all articles published that year. Snapshot of Feb. 2014.

Figure C.13 indicates average citations per article. Sample has median of 10 citations per article, average citations is 51, and standard deviation is 125, which means that the number of citations has high variance among distinct articles.

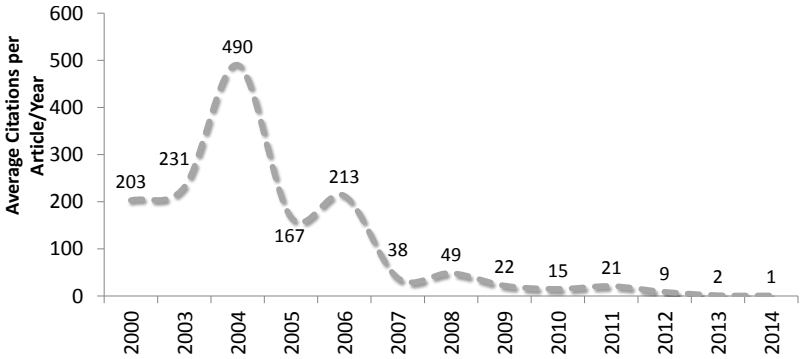


Figure C.13. Average citations per year: x-axis indicates year, while y-axis indicates average citations for all articles published that year. Snapshot of Feb. 2014.

Table C.1 presents citations per classes: 'Architecture' presents median of 13 citations per article. 'Communication' has median of 12 citations per article, while 'deployment' has median of 10 citations.

Table C.1. Comparing citations per classes.

Class	Median	Average	Std dev
Architecture	13	86	181
Communication	12	63	105
Deployment	10	38	95

Figure C.14 presents the histogram of the number of citations per article: less than 10% of the articles surpasses 320 citations.

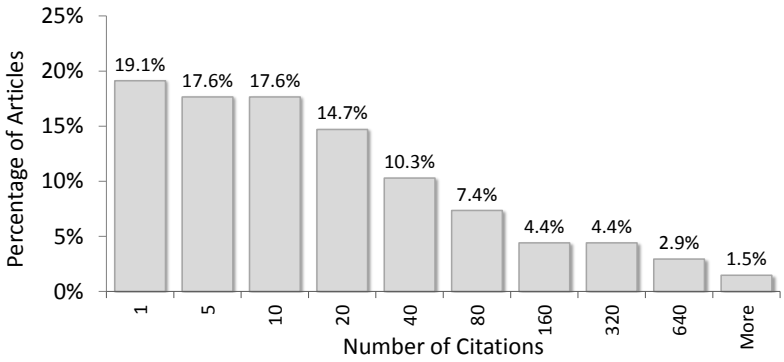
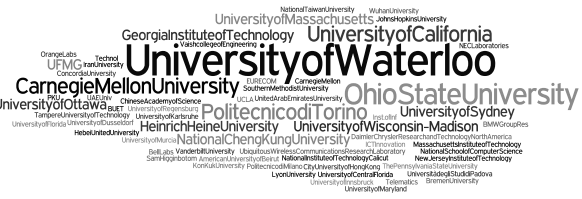


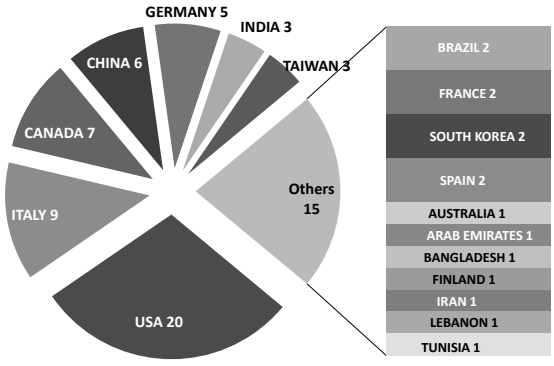
Figure C.14. Histogram presenting Citations per Articles: x-axis indicates number of citations, while y-axis indicates percentage of articles: 19.1% of articles received just 1 citation.

C.4 Characterizing Players: Universities & Labs

In this section we characterize the amount of production generated per university/research lab. We have considered just the affiliation of the first author.



(a) Wordle: Universities and Research Labs.



(b) Number of citations per country.

Figure C.15. Players.

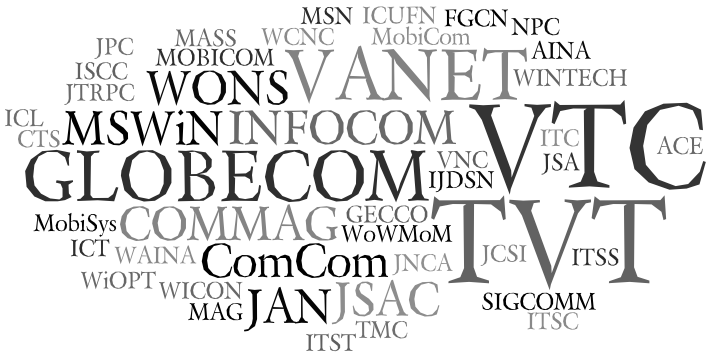
Fig. C.15(a) presents a *wordle* containing the name of each institution proportionally to the number of published articles: University of Waterloo, Carnegie Mellon, Ohio State University, Politecnico di Torino and University of California are

highlighted in Fig. C.15(a). Finally, Fig. C.15(b) presents the number of citations per country. The top-4 players are: USA (2,986 citations), Germany (802), Italy (276) and Canada (225).

C.5 Conferences & Journals

Now we characterize the conferences and journals.

(a) Full Name.



(b) Short Name.

Figure C.16. Conferences and Journals.