Information Dissemination in Mobile Networks

Oscar Trullols Cruces

Advisor: José M^a Barceló Ordinas

Departament d'Arquitectura de Computadors
Universitat Politècnica de Catalunya. BarcelonaTech

A thesis submitted in fulfillment of the requirements for the degree of
Doctor of Philosophy / Doctor per la UPC

— 2014

UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

acknowledgements heading

Acknowledgements / Agraïments

for the acknowlegments

Aquesta tesi no hauria estat possible sense el suport d'un nombrós grup de persones.

En primer lloc vull agrair al meu director de Tesi José Mª, per l'ajut rebut sempre que ha estat necessari, per haver confiat en mi ja des d'abans de començar el màster, i per seguir-ho fent en els nous projectes en els que participa. I també al grup de recerca CompNet en general, per haver posat a la meva disposició tots els recursos que han estat necessaris per portar a terme aquesta tesi.

En segon lloc vull agrair a la meva família, que sense entendre exactament què era el Doctorat i perquè un cop acabada la carrera i el màster seguia a la Universitat, m'han donat el suport per seguir endavant durant tots aquests anys.

Molt agraït també del Marco Fiore, que ha tingut un paper molt rellevant en aquesta tesi, col·laborant i treballant conjuntament, des de que va venir de post-doc a Barcelona, en una gran quantitat de projectes i papers, i amb qui encara ara seguim treballant. I a tots amb els qui he col·laborat durant tots aquests anys, en especial amb els qui més ho he fet Claudio, Carla, i Julián, sense la vostra ajuda, idees i comentaris aquesta tesi no seria tant completa.

També vull donar les gràcies al Gunnar, per l'oportunitat de fer l'estança en el KTH i rebrem en el seu grup de recerca, on vaig aprendre a treballar i a modelar les xarxes de vianants que han completat el que inicialment eren només xarxes vehiculars. I a la gent que allà vaig conèixer i amb qui he anat coincidint en congressos Vladimir i Ljubica.

A tots els amics del Campus Nord, als que vaig conèixer a la FIB ara fa ja una pila d'anys i als que s'han anat afegint durant el doctorat: Marc, Ignasi, Beacco, Albert, Victor, Carlos, Niko, Enric, Javi, Gemma, Manu, Xavi, René, Davide, Enrique,... i una llarga llista de gent que ha passat pel C6-E208 i sales veïnes, gràcies per tots els cafès, dinars, futbolins, partits de volei i futbol(que rarament guanyàvem), excursions, viatges i estones que hem passat junts, que han fet que aquests últims 10 anys hagin passat volant.

Agrair al Robert i a tots els amics de Karate de l'Equip4, de qui he aprés que per superar amb èxit qualsevol objectiu, ja siguin Matagalls-Montserrats o fer un tesi, és molt important anar pas a pas i estar sempre ben acompanyat.

No puc acabar sense donar les gracies també a tots els amics de Sant Sadurní, que m'heu ajudat a desconnectar durant aquests anys, i que teníeu ja, les mateixes ganes que jo, o més, de que acabés la tesi.

Finalment, un agraïment especial a la Glòria. Que és qui ha compartit amb mi totes les alegries i celebracions que cada pas de la tesi anava donant, però també ha hagut de patir de més a prop que cap altre, les meves èpoques d'estrès per acabar treballs a temps i interminables assaigs de presentacions.

I per acabar, moltes gràcies per la paciència que tots heu demostrat, per a que pogués acabar la tesi al meu ritme...

Aquesta Tesi ha estat realitzada amb el suport econòmic de la Generalitat de Catalunya, mitjançant un contracte predoctoral per a la formació de personal investigador novell FI-DGR amb número d'expedient 2011FI-B2 00019

Abstract

This thesis proposes some solutions to relieve, using Wi-Fi wireless networks, the data consumption of cellular networks using cooperation between nodes, studies how to make a good deployment of access points to optimize the dissemination of contents, analyzes some mechanisms to reduce the nodes' power consumption during data dissemination in opportunistic networks, as well as explores some of the risks that arise in these networks.

Among the applications that are being discussed for data off-loading from cellular networks, we can find **Information Dissemination in Mobile Networks**.

In particular, for this thesis, the Mobile Networks will consist of Vehicular Ad-hoc Networks and Pedestrian Ad-Hoc Networks. In both scenarios we will find applications with the purpose of vehicle-to-vehicle or pedestrian-to-pedestrian Information dissemination, as well as vehicle-to-infrastructure or pedestrian-to-infrastructure Information dissemination. We will see how both scenarios (vehicular and pedestrian) share many characteristics, while on the other hand some differences make them unique, and therefore requiring of specific solutions. For example, large car batteries relegate power saving techniques to a second place, while power-saving techniques and its effects to network performance is a really relevant issue in Pedestrian networks.

While Cellular Networks offer geographically full-coverage, in opportunistic Wi-Fi wireless solutions the short-range non-full-coverage paradigm as well as the high mobility of the nodes requires different network abstractions like **opportunistic networking**, **Disruptive/Delay Tolerant Networks (DTN)** and **Network Coding** to analyze them.

And as a particular application of Dissemination in Mobile Networks, we will study the malware spread in Mobile Networks. Even though it relies on similar spreading mechanisms, we will see how it entails a different perspective on Dissemination.

Contents

1

Introduction

Since the beginning of Internet, the network has evolved in many ways. From the initial research into packet switching, in the early 1960s, until now, many improvements have moved packet switching communications from linking two Research Labs in 1969, to link millions of Personal Computers. At first, linking a small network of laboratories was a challenge hard enough, but after a while, researchers found solutions that allowed the Network to spread far beyond its limits.

In 1991 the precursor to Wi-Fi was invented and we can find the first 802.11 Wi-Fi patents from 1996. This was the first step to make Internet available without wires, you just had to be close to an Access Point to connect to the Internet at high bandwidths.

In 2000, people were used to Internet, and it began the interest in accessing it from "anywhere" at anytime. For example, in some use cases, people started using it to check the email from an airport Wi-Fi hotspot, read on-line newspapers at the library, or write a post for a blog from a coffee shop.

Nowadays their needs and requirements have increased, users want to find directions or use on-line maps in a city, read the last minute news during a bus trip, find restaurants and facilities around their current position, instant messaging, gaming, on-line radios, and a lot of services still under development.

Internet has moved from bulky machines used to share resources among research labs, to the pocket of many users around the globe in the form of small but powerful hand-held devices ("smart-phones").

And the solutions that make this possible, are based on different technologies and mechanisms. Among them, it is of particular interest for this thesis the "final wire-

less hop" and the new applications it raises. In particular, we are interested in Wi-Fi solutions, without despising or forgetting Cellular Network solutions, but focusing our interest in Wi-Fi solutions with their features, assuming its advantages and disadvantages.

The penetration rate of Wi-Fi powered devices in the users pockets and vehicles is high and increasing, we are not talking about future applications and services anymore, the interest exists and we can find people using them on the streets.

And there are no reasons to believe this will stop here. New technological advances encourage new user applications (e.g Smart Cities, Internet of Things, Big Data ...), and the new user interests pressure technology to go one step further (e.g Delay Tolerant Networks (DTN), Pedestrian Networks, Vehicular Ad hoc Networks (VANETS), Sensor Networks, ...).

In particular, Mobile Ad hoc Networks, including VANETS, Pedestrian Networks and DTN is the topic we have chosen for our research. And one of the Mobile scenarios where we have dedicated most of our efforts are Vehicular Ad-hoc Networks, whereas many of the proposed ideas are applicable or easily ported to other Mobile scenarios.

Some of the applications we have mentioned have been implemented or could be implemented by using 3G/4G cellular network, but in our thesis we think it is justified and even essential the research in Wi-Fi solutions for this scenario, because of the benefits it can pose. In some cases, wireless technology can be seen as complementary to the 3G/4G network (to lighten the load on the cellular network, or enhance the service where cellular coverage is low), while in other cases the Wi-Fi technology characteristics can offer some benefits not provided by other solutions, for its immediacy (e.g. emergency braking, extended vision on crossroads), geographical interest (e.g. real time traffic maps, free places to park), bandwidth (audio/video announcements, passenger-infotainment), robustness based on the non-centralized approach, etc.

And to conclude this part of the introduction, with some motivation based on recent data about the Mobile Internet consumption scene, and its growth:

- "Last year's (2012) **Mobile data traffic** was nearly twelve times the size of the **entire global Internet** in 2000", "reaching 885 petabytes per month at the end of 2012" (1).

- "Average smartphone usage grew 81 percent in 2012. The average amount of traffic per smart-phone in 2012 was 342 MB per month, up from 189 MB per month in 2011" (1).

The users have shown an increasing consumption of this type of data, and it relies on the technology to continue supplying this increased need for resources. Currently, the reports show that a large portion of the mobile total network data consumption is downloaded from Wi-Fi networks.

- "Globally, 33 percent of total mobile data traffic was offloaded onto the fixed network through Wi-Fi" (1).

Thus relieving the burden on the 2G/3G network from Wi-Fi technologies. These facts support the claim that both technologies are complementary, and must cooperate to maintain customer QoS requirements at lower costs for the users and the operators.

The same reports claim that Cellular Operators have already encouraged users to offload the traffic onto Wi-Fi networks:

- "Since first introduced in 2009 and 2010, the majority of mobile users have now been migrated to tiered plans. Many operators across the globe have eliminated unlimited data plans" (1).

- "Operators have encouraged the offload of traffic onto Wi-Fi networks, and offload rates continue to be high around the world. Tablet traffic that might have migrated to mobile networks has largely remained on fixed networks" (1).

The next sections introduce some of the topics presented in this thesis.

1.1 Mobile Ad Hoc Networks

Mobile Ad Hoc Networks (MANETs) are characterized as networks in which nodes, static or mobile, act as a host **and** as a router extending the one-hop coverage area of a single wireless network. These networks are self-organized, and typically, nodes follow random mobility patterns.

MANETs may be impacted by several factors:

- High speed of the nodes.

- Environmental factors.
- Determined mobility patterns and street/road/building conditions.
- Intermittent communications (isolated nodes or small clusters due to the fragmentation of the network).
- High congestion channels (e.g. due to high density of nodes).

Packets traverse an ad hoc network by being relayed from one node to another until they reach their destination. Because nodes are moving, the topology of the network is in constant change, and to find the route and destination could be a hard challenge. Routing in mobile ad hoc networks is a well-studied topic, several routing protocols have been proposed, such as OLSR (2), AODV (3), DSR(4), while (5) reviews routing protocols for mobile ad hoc networks.

These solutions are very useful in some kind of ad hoc networks, but they would not work under our proposed scenarios because they incorrectly assume the existence of:

- An end-to-end path between any pair of nodes.
- Small maximum round-trip time between them.
- Small hop-by-hop and end-to-end packet drop probability.

Mobile Ad Hoc Network consider all kind of Mobile nodes, but in our case, we base our work mainly on **vehicles** and **pedestrians**, and we will see these in more depth next:

1.1.1 Vehicular Ad-hoc Networks

Vehicular Ad-hoc NETworks (VANETs) are a particular case of Ad Hoc networks in which nodes are vehicles that move following specific patterns (i.e., roads, highways, streets) and follow vehicle traffic regulations. VANETs are networks characterized by intermittent connectivity and rapid changes in their topology. In contrast with other ad-hoc networks, these networks also have very specific mobility patterns due to vehicle traffic signals. Different from traditional data access system in which users can always wait for the service from the data server, vehicles are moving and they only stay in the

Road-Side Units (RSUs) areas for a short period of time. Meanwhile, to make the best use of the RSUs, they are often set at areas with high traffic. In these areas requests compete for the same limited bandwidth. As an example consider vehicles traveling on highways that have sparse RSUs distributed along the route. Due to the high speeds, vehicles have few seconds to access Internet or other vehicles to which may want to communicate. Furthermore, the environment presents a high level of packet losses. For example, measurements of UDP and TCP transmissions of vehicles in a highway passing in front of an Access Point(AP) moving at different speeds, report losses on the order of 50-60% depending on the nominal sending rate and vehicle speed, (6). In ranges of around 250 meters, throughput reaches approximately 4 Mb/s, while at larger distances (e.g. 400 meters) the throughput drops to around 1 Mb/s.

It should be pointed out that vehicular networks share, and possibly exacerbate, the typical shortcomings of ad hoc networks. Specifically: fleeting connectivity, rapidly shifting topologies, highly dynamic traffic patterns, constrained node movements. In particular, unlike cellular communication networks, vehicular networks do not necessarily need continuous coverage, rather, they can be supported by hot spots in correspondence of roadside infrastructure nodes, which provide intermittent connectivity to vehicles. The challenges featured by this scenario are therefore more related to the ones typically found in DTN (see section 1.2.1 Disruption-Tolerant Networks) (7, 8) than in infrastructure-based wireless networks.

Vehicles traveling within cities and along highways are commonly regarded as most probable candidates for a complete integration into mobile networks of the next generation. **Vehicle-to-infrastructure (V2I)** or **Vehicle-to-Road (V2R)** and **vehicle-to-vehicle (V2V)** communication could indeed foster a number of new applications of notable interest and critical importance, ranging from danger warning to traffic congestion avoidance. It is however easy to foresee that the availability of on-board communication capabilities will also determine a significant increase in the number of mobile users regularly employing business and infotainment applications during their displacements. As a matter of fact, equipping vehicles with WiMAX/LTE and/or Wi-Fi capabilities would represent a clear invitation for passengers on cars or buses to behave exactly as home-based network users. The phenomenon would thus affect not only lightweight services such as web browsing or e-mailing, but also resource-intensive ones such as streaming or file sharing.

Although this could not represent a problem for relatively lightweight services such as web browsing or e-mailing, resource-intensive tasks such as video streaming or file sharing will instead risk to overload the wireless communication infrastructure. This could result in much worse breakdowns than those already faced today by cellular networks in front of the growing number of high-end mobile users.

In order to support these demanding operations and thus favor the network scalability, a valuable aid to the traditional user-to-infrastructure communication paradigm could come from interactions among mobile users. Within such context, the fast movement dynamics that characterize vehicular environments make fully ad-hoc approaches, that try to build a connected network over moving cars, impractical. Instead, opportunistic vehicle-to-vehicle communication appears as a more viable complement to infrastructure-based connectivity.

1.1.2 Pedestrian Ad-hoc Networks

During the last years, the mobile wireless capabilities of portable devices (e.g smartphones, multimedia players) have increased. It is not one but several, the enhancements that have been introduced before reaching its current massive penetration in the population.

Improvements in wireless communications, batteries and technological advances made more energy efficient devices, allowing users to have them in their pockets, switched on and fully operative for more than a full day long.

In the case of smartphones, while their main wireless technology is the Cellular Network, those devices are also equipped with Wi-Fi technology, and Pedestrian Ad-hoc Networks study specifically the characteristics, features, mechanisms and applications that show up on this new framework.

In comparison with the Cellular Networks, the shorter range of Wi-Fi offers higher spatial reuse for larger amounts of users, resulting in a higher bandwidth per user. By not depending on the operator's infrastructure, those networks may not require a monthly fee, as they can be self supported by the investment of each of its members in his own device. But this same reason pose an increased complexity for the network ad hoc organization and management.

Pedestrian Ad-hoc Networks, being a subgroup of the MANETs, inherit most of the characteristics already presented, while add or make some them more stringent:

- Scarce battery.
- Scenario and mobility heterogeneity (e.g. street, mall, stadium, park, indoors,...).
- Technology heterogeneity (e.g. Wi-Fi, Bluetooth).
- Non dedicated resources, ISM band.

Until recently, most of the research in MANET and Pedestrian networks, has been conducted assuming simple and homogeneous Human Mobility (e.g Random Walks, Random Waypoints), while Human Mobility is a very complex process, and difficult to reproduce with analytical models. While Random Walks or Random Waypoints may fit some concrete scenarios, it has been shown that they can also produce very unrealistic scenarios, that tend to overestimate or underestimate the performance of some solutions.

1.2 Network Paradigms

A common scenario in opportunistic networking is a geographical area with no Wi-Fi infrastructure or with sparse Wi-Fi access points (AP). Applications in this scenario range from peer-to-peer message exchange, publish/subscribe applications or shops that want to publish marketing products to potential customers.

Users move around the area and obtain targeted information when they cross the hot spots, or users want to communicate some data to other users. However, when they leave the coverage area of the transmission, the wireless link is lost and the transmission is disrupted. If the information is not time-sensitive, connectivity can be recovered when the user again meets a Wi-Fi access point or when the user finds a new useful contact opportunity.

This kind of networks are called **Delay/Disruptive Tolerant Networks** (9), and posed a new networking paradigm in the area of wireless communications. Another Network Paradigm which offers great benefits to opportunistic wireless networks with packet loss is **Network Coding (NC)**. While NC can be used with different purposes in other scenarios, in our case this paradigm offers an alternative to forward error correction and ARQ mechanisms.

1.2.1 Delay/Disruption Tolerant Networks

Delay/Disruption Tolerant Networking (DTNs), (9), are wireless mobile networks that use intermittently available links to communicate opportunistically, using a store-carry-and-forward paradigm. DTNs are particularly useful in sparse environments where the density of nodes is insufficient to support direct end-to-end communication. The main performance metrics of a DTN are deliverability and delay, which are critically dependent on the node mobility patterns that drive the frequency, duration and sequence of contact opportunities. Moreover, DTN nodes are usually untethered devices with limited energy supplies, thus making energy consumption a primary concern, in particular the energy consumed in searching for other nodes to communicate with.

In a mobile DTN, two nodes communicate with each other during the *contacts* that occur when both nodes, either mobile or stationary, are within the radio range of one another. On the other hand, the wireless interface is one of the largest energy consumers in mobile devices, whether they are actively communicating or just listening (10), which means that there is a clear trade-off between saving energy and providing connectivity through opportunistic encounters.

1.2.2 Network Coding

Network coding allows efficient transmission from a set of sources to a set of destinations, allowing nodes to manipulate the information before forwarding it (11). It has been originally employed to improve the throughput of multicast transmissions over reliable links (11). Random linear network coding is a class of network coding, that operates on data through linear combinations of random codes (12). Although initially employed for multicast, random linear network coding has found wide application in networks with packet erasures, where it is used to improve the communication performance in absence of (or with limited resort to) feedback channels. Random linear network coding has been shown to improve the latency, capacity and energy efficiency of the communication in loss-prone and intermittently-connected wireless networks, either ad-hoc (13), delay-tolerant (14), or satellite and underwater (15).

1.3 Applications: Data Dissemination

One of the most interesting applications in opportunistic networks is the dissemination of information in which an application (e.g. could be a pub/subs application or a marketing application) delivers a delay tolerant object to the network with the purpose of being accessed by as many users as possible.

The concept is very similar to an infection system in which a person who suffers a viral disease, transmits the infection from individual to individual among the rest of the population, (16). Papers (17) or (18) apply epidemic SIR (Susceptible-Infectious-Recovered) modeling to opportunistic networking in which users meet each other and opportunistically exchange data.

There are many cases where Data Dissemination applications are applied to MANETS, or specifically to VANETS (19) , (20) , (21) , (22) , (23), , (24).

Data Dissemination applications are designed with the following aims:

- Reduce the overhead of the protocols.
- Optimize the data replication and redundancy to achieve high delivery ratios with low network resources consumption.
- Fair sharing of the resources among users and broadcasters.

While in Data Dissemination applications the content being broadcast can be any given data, there are more specific cases, where while the propagation and spreading follows similar characteristics the purpose of the application is completely different and not all the previous aims apply, or attempt just the opposite, this is the case of **malware in MANETS**.

1.3.1 Malware in MANETS

Pervasive wireless machine-to-machine (M2M) communication is foreseen to be a game changer for many daily life activities, other than a technology enabling a broad range of new applications. This motivates the ever-growing availability of a wide variety of long- and short-range communication-enabled devices, from smart-phones to tablets, from notebooks to microwaves, from refrigerators to cars. The new generation of smart

objects shall grant faster, cheaper communication with our friends and co-workers, easier home management, safer and more efficient mobility.

However, as it is often the case, with great profits comes high risk. If not properly secured, the network interfaces of smart devices can turn into easily exploitable backdoors, allowing illegal remote access to the information stored on the device as well as to the local network it may be connected to. Even worse, M2M communication could be leveraged by self-propagating malware to reach a large number of devices and damage them, disrupt their services or steal sensible data. The first mobile malware that spread itself through Bluetooth wireless connection, the *Cabir* worm (25), appeared in 2004 and was soon followed by several evolutions (26, 27).

The low penetration of smart devices and the heterogeneity of the operating systems have prevented major outbreaks of M2M-based worms to date (28). However, as the diffusion of communication interfaces keeps growing and the OS market becomes more stable, with two or three major competitors remaining, it is easy to predict that we may have to face smart-device worm epidemics in the future. It is thus important to understand today which are the risks we may be facing tomorrow. In fact, the behavior of potential malware in different sensible M2M communication environments has been a subject of research ever since Cabir made its first appearance. Simulative and experimental studies have outlined the risks yielded by the diffusion of so-called mobile worms via direct Bluetooth infection, both within campuses (29) and in urban areas (30), via metropolitan Wi-Fi hot-spot associations (31), and through text messaging in cellular networks (32). In all these cases, it was found that, although mobile worms propagate at speeds that are orders of magnitude lower than their Internet counterparts, they are less easily detectable and still fast enough to pose a threat.

One of the scenarios where mobile malware could cause the most damage is the automotive one. Indeed, vehicles feature today a wide range of Electronic Control Units (ECUs) interconnected by a bus, e.g., the Controller Area Network (CAN), that directly determine most of the cars' automatic behaviors. Experimental tests have proven that not only ECUs are extremely fragile to the injection of non-compliant random messages over the CAN, but that a knowledgeable adversary can exploit them to bypass the driver input and take control over key automotive functions, such as disabling brakes or stopping the car engine (33).

Lives could be thus put at stake, if a remote attack was run against a moving vehicle's ECUs. What is worse is that the above has been proved to be feasible even remotely, by exploiting the Tire Pressure Monitoring Systems (TPMS) (34) or the CD player, Bluetooth and cellular interfaces (35).

And, in that sense, the forthcoming IEEE 802.11p-based WAVE interfaces, allowing direct vehicle-to-vehicle (V2V) communication, risk to significantly widen the range of attack surfaces available to adversaries.

1.4 Mobility Models and Methodology

While the purpose of this thesis is not to develop new Mobility Models or to improve current mobility methodology, our goal has always been to use the state of the art Mobility Models, Mobility Simulators, and the best Mobility traces available we have been allowed access.

That said, it's also needed to mention that many times during this thesis, we have been working closely with other researchers, with extensive experience in Mobility Modeling Methodology, whose main aim in their research was to develop and polish models, simulators and traces for vehicular or pedestrian simulations.

Some examples of these collaborations with other authors, where we have combined their efforts in Mobility Modeling with our efforts in Dissemination or Cooperation algorithms, to learn together about the effects of more realistic Mobility Models in protocols and applications performance, are publications like:

- **Generation and Analysis of a Large-scale Urban Vehicular Mobility Dataset** (36) S. Uppoor, O. Trullols Cruces, J.M. Barceló and M. Fiore,
- **VANET Mobility Modeling Challenged by Feedback Loops** (37) H. Meyer, O. Trullols Cruces, A. Hess, J.M. Barceló, K.A. Hummel, C.E. Casetti and G. Karlsson
- **Vehicular Networks on Two Madrid Highways** (38) M. Gramaglia, O. Trullols Cruces, D. Naboulsi, M. Fiore and M. Calderon

Therefore, although the Mobility Models are not the focus of this thesis, it will be shown how they have, on the simulation results, a major effect that can not be despised.

Not only that, but also our results have benefited from our involvement in these joint projects, the advice we have been given from the people we have worked with, the knowledge they shared and the tools and resources they had already gathered.

Each of the following chapters describes the specific mobility methodology that has been used for the different scenarios.

1.4.1 Generic Mobility Models

The performance evaluation of any Network Protocol or Application requires as realistic conditions as possible. Therefore, Mobile Networks performance evaluation, among other things, require realistic Mobility Models.

As there are many different Mobile Networks scenarios, the best choice for each may differ. And while the more complex mobility models capture some features and characteristics from the real scenarios better, many papers in the literature have used a well known set of synthetic Mobility Models. T. Camp et al. in (39) presents a survey of Mobility models for ad hoc networks research. It reviews the most used synthetic Mobility Models:

- **Random Walk (RW)** This simple mobility model is based on giving mobile nodes random directions and speeds. Some of the following models can be considered as refinements of this one, as there are many variations based on when nodes change direction and speed: after a (i)fixed or (ii)random time, or after a (iii)fixed or (iv)random distance. Other Random Walk choices are the node's behaviour on reaching the boundaries: (i) new random values or (ii) direction reflexion.

- **Random Waypoint (RWP)** In this model, the node picks a random destination and speed and travels there in a straight line. It may include random/fixed pause times between changes.

- **Random Direction (RD)** This model forces mobile nodes to travel to the edge of the simulation area before changing direction and speed.

- **A Boundless Simulation Area** Random Walk variation that converts a 2D rectangular simulation area into a torus-shaped simulation area.

- **Gauss-Markov** The model uses one tuning parameter to vary the degree of randomness in the mobility pattern.

- **A Probabilistic Random Walk** A model that utilizes a set of probabilities to determine the next position of an MN.

- **City Section and Manhattan Mobility Model** A simulation area that represents streets within a city, nodes can only move on the roads/streets, traveling to the random destination following the shortest path.

1.4.2 Pedestrian Mobility Models

The same Mobility models presented for generic Mobile Networks, have also been widely used in Pedestrian Networks' research. And it is to this scenario, where some works like (40) and (41) introduce more complex pedestrian behaviours, route choice and activity scheduling in their models. Those works compare their outputs with the previous Models, and validate proposals with real statistical patterns.

The paper (42) presents and discusses a pedestrian Mobility Model that is part of the freely available UDel Models (43), a suite of tools for realistic simulation of urban wireless networks. The model uses an extensive set of surveys from the US Bureau of Labor Statistics for its calibration and validation.

1.4.3 Vehicular Mobility Models

Similar works have been carried out in Vehicular Mobility Models. A long and detailed review of Vehicular Mobility Models can be found in (44), where J. Harri et al. present a survey and a taxonomy of Mobility Models for Vehicular Ad Hoc networks. Another interesting work on the topic is (45), where the authors analyze the impact of the Mobility Model on the network connectivity metrics.

Vehicular Mobility Models are often classified as **microscopic** or **macroscopic**.

- The **microscopic** models consider individual driver behavior and dynamics, and her reactions to the near vehicles. These include fine-grained real world situations, as overtakings, brakings, speed adaptations, etc.

- The **macroscopic** models consider vehicular traffic as a continuous flow with generic metrics of interest. These include traffic density, traffic flows, average speeds, etc.

While the first Vehicular Mobility Models focused on micro **or** macro perspectives, current Models and simulators target both with a high level of realism (36).

1.5 Rationale

Information Dissemination in Mobile Networks is a novel and very wide topic, with many open problems (some of them may have not even been raised yet.) Many issues remain to be defined, discussed and solved. We think that many key points need to be addressed before new generic applications can be delivered to the end user, there are many things to improve and problems to solve.

Our work on Information Dissemination in Mobile Networks begun during my Master Thesis with the study on cooperation in VANETs . We started working with Julián Morillo on Cooperative ARQ(Automatic repeat request) algorithms, applying his thesis (46) results to the Vehicular scenario (47), (48), (49).

In his work, the cooperation resides in the Data Link Layer of the OSI model, while we were also interested on Cooperation at higher network layers. And specifically on cooperation in **Information Dissemination in Mobile Networks**, that gives title to this thesis.

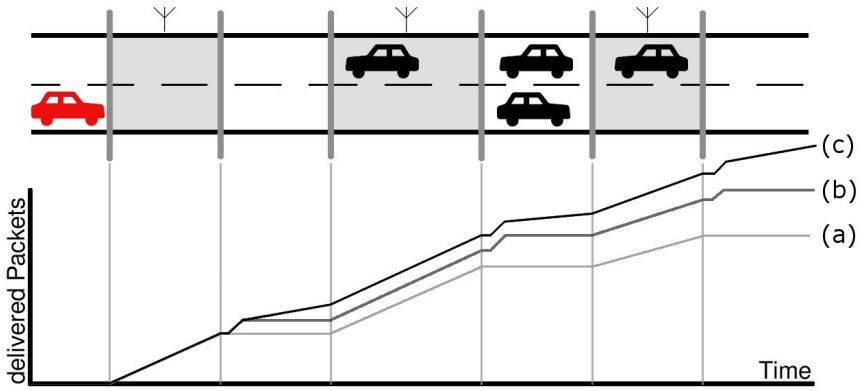


Figure 1.1: cooperation in Information Dissemination

Figure 1.1, from our joint work in (50), shows briefly how this first cooperation with Julian, mixes both cooperative ARQ and opportunistic contacts to improve the Data dissemination in a Highway scenario. Line (a) shows the evolution over time of the delivered Packets to a vehicle along a highway without using cooperation. Line(b) shows the behaviour expected from cooperative ARQ: after the nodes leave the Access Point coverage, they cooperate to recover missed packets that some of their neighbors may have correctly received, while line (c) shows the expected behaviour using both Cooperative ARQ and opportunistic contacts: nodes keep receiving packets from cooperators while they travel between Access Points. Our next step was to move to a more complex scenario, where one-dimensional highway movements give way to the more complex urban scenarios presented in (51) and chapter 3 of this Thesis.

While this part of the thesis focuses on **Unicast Information Dissemination** (data has a unique destination), in parallel we focused on **Broadcast Information Dissemination** (data needs to be delivered to every node), and in this field we faced several issues: how to improve the deployment of Road side units to optimize the Information Dissemination (52), (53), (54), (55) (Chapter 5), how the information spreads in a large urban scenario (56) (Chapter 6)(using as an application the spread of a Malware), and how Power Saving techniques impacts the Information spread in Pedestrian Networks (57),(58) (Chapter 7).

Moreover, while we were working on these Information Dissemination scenarios, we considered to apply Random Linear Network Coding to improve the reliability in Mobile Networks. And while we were learning how to use it, we found that an upper bound was being used for the decoding probability, while we thought that the exact formulation could be found (59) (Chapter 4)

During this thesis we have dedicated our efforts on the points we thought were more needed to make Information Dissemination in MANETS possible. Figure 1.2 shows a diagram of the different research areas related with the topic of this thesis. In the figure, letters A-E show the points where we have focused and how they relate among themselves. In this section we list the hypothesis and purposes of the thesis for each of these areas:

- **A. VANETS can benefit from vehicular cooperation** to improve the performance of the Mobile Ad hoc Network.

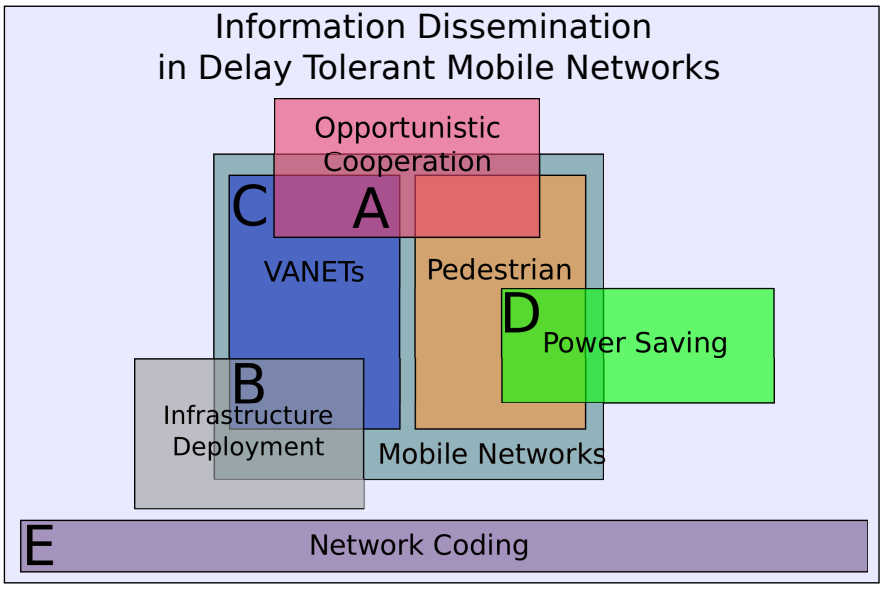


Figure 1.2: issues addressed in this thesis.

We propose to use simple mechanisms to reuse its underutilized resources, as well as currently unexploited V2V opportunities.

Our works related with this purpose have been gathered in the Summary of original work section **2.2 Opportunistic Cooperation**.

- **B. Using statistical data of the Vehicular Mobility it is possible to enhance cooperation, as well as improve RSU deployments in city wide scenarios.**

  The more informed the algorithms used to plan RSU deployments the better tuning of the system can be achieved, while just some aggregated information or statistics about Vehicular Mobility in cities and urban or interurban scenarios can help to achieve RSU deployments that would disseminate effectively data contents to most of the vehicles as fast as possible.

  We propose to study the possibilities that gathering vehicular mobility data offers, how can it be used to forecast vehicular cooperations, and how to use it to deploy a limited number of RSUs under economic budget restrictions.

  Our works related with this purpose have been gathered in the Summary of original work section **2.3 Roadside Infrastructure Deployment for Information**

Dissemination.

- **C. Malware in VANETS can spread rapidly, and pose a real threat**, while malware threats are still not studied in this scenario, we believe it can pose a real threat that can spread to large parts of cities in a short time. This scenario is difficult to simulate, as mobility traces are not available for every city and are costly to obtain. While, on the other hand Transportation authorities and Governments already have some per-road statistics.

  We propose to study this scenario, using the few available large scale vehicular mobility traces, to study them and obtain an analytical model based on the per-road statistics gathered by transportation authorities that mimics the vehicular traces, and can be used to ease the burden studying this kind of scenarios in other cities as well as under different malware characteristics.

  Our works related with this purpose have been gathered in the Summary of original work section **2.4 Malware in MANETS**.

- **D. There is a clear trade-off between the power that can be saved using power saving techniques in MANET nodes and the contact opportunities missed in this kind of networks**. Some battery powered devices, use On/Off duty-cycle techniques to save power. During the Off periods, the devices may miss contact opportunities that are so important in MANETS.

  Our purpose is to study and define the trade-off relation between energy saved and the missed contact opportunities in MANET scenarios. To allow others to tune their frameworks to save power while considering the effects on other network performance metrics.

  Our works related with this purpose have been gathered in the Summary of original work section **2.5 MANET Power Saving Trade-offs**.

- **E. Some of the currently used decoding probability error bounds used in Network Coding can be improved.** Network Coding is used in many frameworks and projects, but as this decoding probability error is small when using NC under large Galois Fields, many works so far, neglect it or use the approximated upper bound error to build their frameworks.

We propose to study the decoding error probability, to find the exact formulation to find this value. To define the number of extra packets that must be sent as a function of the number of packets and the Galois field used.

Our works related with this purpose have been gathered in the Summary of original work section **2.6 Analysis of Random Linear Network Coding**.

1.6 Outline of the Document

This thesis is organized as follows:

A summary of the original work published in the context of this thesis and my specific contributions to each of these works is provided in Chapter 2. Chapter 3 addresses Unicast Information Dissemination or Cooperative Download in Vehicular Environments, is an adaptation of paper (51). Chapter 4 shows how to calculate the exact decoding probability under Random Linear Network Coding, is an adaptation of paper (59). The next three chapters address Multicast Information Dissemination, Chapter 5 focuses on planning roadside infrastructure deployment, is an adaptation of paper (54). Chapter 6 focuses on the characterization and modeling of the Information Dissemination spread in a large urban scenario, is an adaptation of paper (56). While Chapter 7 closes the Multicast Information Dissemination part, focusing on the dissemination of information on Pedestrian Networks under Power Saving Constrains, is an adaptation of paper (57). Finally, General Conclusions are given in Chapter 8

2

Summary of Original Work

This chapter starts by listing all the publications derived from the work done during the period of this thesis.

Some publications have been written by joining efforts with other authors. In the second part of the chapter, I have tried to delimit my contributions to each of these works. Although, when working in group sometimes it is not clear or fair to claim authorship. Specially when working in small groups (with my PhD Advisor J.M. Barcelo and M. Fiore) everybody has been involved with everything. In other cases, when collaborating in projects with larger groups, each member of the project has contributed with part of it and it has been easier to be more accurate with my contribution's description.

2.1 List of publications

Opportunistic Cooperation

Cooperative Download in Vehicular Environments,
O. Trullols-Cruces, M. Fiore, J.M. Barcelo-Ordinas,
IEEE Transactions on Mobile Computing (TMC), Issue 99, May 2011,
Impact Factor or CORE rank: 2.647, A* Cited by: 16[1]

A Cooperative Vehicular Network Framework ,
O. Trullols-Cruces, J. Morillo-Pozo, J.M. Barcelo-Ordinas, J. Garcia-Vidal,
IEEE International Conference on Communications (ICC), Dresden, Germany, June 2009,
Impact Factor or CORE rank: A Cited by: 18[1]

Applying Cooperation for Delay Tolerant Vehicular Networks,
J. Morillo-Pozo, J. M. Barcelo-Ordinas, **O. Trullols-Cruces**, J. Garcia-Vidal,
4th EuroFGI Workshop on Wireless and Mobility, Barcelona, Spain, September 2008,
Impact Factor or CORE rank: - Cited by: 7[1]

Evaluation of a Cooperative ARQ Protocol for Delay-Tolerant Vehicular Networks,
J. Morillo-Pozo, J. M. Barcelo-Ordinas, **O. Trullols-Cruces**, J. Garcia-Vidal,
Lecture Notes in Computer Science (LNCS), Volume 4396, pp 43-61, July 2008,
Impact Factor or CORE rank: 0.4 Cited by: 4[1]

A Cooperative ARQ for Delay-Tolerant Vehicular Networks,
J. Morillo-Pozo, J. M. Barcelo-Ordinas, **O. Trullols-Cruces**, J. Garcia-Vidal,
IEEE International Workshop on Delay-Tolerant Mobile Networks (DTMN'08), June 2008,
Impact Factor or CORE rank: A Cited by: 18[1]

Roadside Infrastructure Deployment for Information Dissemination

RSU Deployment for Content Dissemination and Downloading in Intelligent Transportation Systems,
M. Reineri, C. Casetti, C.F. Chiasserini, M. Fiore, **O. Trullols-Cruces** and J.M. Barcelo,
Book Chapter to "Roadside Networks for Vehicular Communications", 2013,
Impact Factor or CORE rank: - Cited by: 1[1]

Planning Roadside Infrastructure for Information Dissemination in Intelligent Transportation Systems ,
O. Trullols-Cruces, M. Fiore, C. Casetti, C.F. Chiasserini, J.M. Barcelo-Ordinas,
Computer Communications, Elsevier, January 2010,
Impact Factor or CORE rank: 0.933 Cited by: 45[1]

A Max Coverage Formulation for Information Dissemination in Vehicular Networks,
O. Trullols-Cruces, M. Fiore, C. Casetti, C.F. Chiasserini and J.M. Barcelo,
IEEE Int. Conf. on Wireless and Mobile Computing, Networking and Communications (WIMOB), October 2009,
Impact Factor or CORE rank: C Cited by: 4[1]

[1]Google scholar citations April 2014

Analysis of Random Linear Network Coding

Exact Decoding Probability Under Random Linear Network Coding ,
O. Trullols-Cruces, J.M. Barcelo-Ordinas, M. Fiore,
IEEE Communication Letters, January 2011, ISSN:1089-7798, (online)November 2010,
Impact Factor or CORE rank: 1.140 Cited by: 30[1]

Information Dissemination

Understanding, Modeling and Taming Mobile Malware Epidemics in a Large-scale Vehicular Network ,
O. Trullols-Cruces, M. Fiore, J.M. Barcelo-Ordinas,
IEEE 14th World of Wireless, Mobile and Multimedia Networks (WoWMoM), Madrid, Spain, 2013,
Impact Factor or CORE rank: A Cited by: 0[1]

Power Saving Trade-offs in Delay/Disruptive Tolerant Networks ,
O. Trullols-Cruces, J. Morillo, J. M. Barcelo and J. Garcia-Vidal,
IEEE 12th World of Wireless, Mobile and Multimedia Networks (WoWMoM), Lucca, Italy, June 2011,
Impact Factor or CORE rank: A Cited by: 6[1]

Impact of the Infrastructure in Mobile Opportunistic Networks(Invited Paper) ,
O. Trullols-Cruces and J. Morillo and J. M. Barcelo,
Int. Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL), October 2011
Impact Factor or CORE rank: - Cited by: 5[1]

Mobility Models

Vehicular Networks on Two Madrid Highways,
M. Gramaglia, **O. Trullols-Cruces**, D. Naboulsi, M. Fiore and M. Calderon,
Accepted for IEEE International Conference on Sensing, Communication and Networking (SECON), 30 June 2014,
Impact Factor or CORE rank: - To be Published

Generation and Analysis of a Large-scale Urban Vehicular Mobility Dataset ,
S. Uppoor, **O. Trullols-Cruces**, M. Fiore, J.M. Barcelo-Ordinas,
IEEE Transactions on Mobile Computing (TMC), february 2013 (online, preprint),
Impact Factor or CORE rank: 2.647, A* Cited by: 8[1]

VANET Mobility Modeling Challenged by Feedback Loops (Invited Paper) ,
H. Meyer, **O. Trullols-Cruces**, A. Hess, K.A. Hummel, J.M. Barcelo, C.E. Casetti and G. Karlsson,
10th IEEE/IFIP Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), June 2011,
Impact Factor or CORE rank: - Cited by: 5[1]

[1]Google scholar citations April 2014

2.2 Opportunistic Cooperation

2.2.1 A Cooperative Vehicular Network Framework (50)

O. Trullols-Cruces, J. Morillo-Pozo, J.M. Barcelo-Ordinas, J. Garcia-Vidal

IEEE International Conference on Communications, Dresden, Germany, June 2009

Impact Factor or CORE rank: A Cited by: 18 [1]

Contribution: This paper extends and presents some of the most interesting results obtained during my Master Thesis. My contribution to this work has been the problem formulation, the design, implementation and evaluation of the framework under the supervision and guidance of the other authors. The design and prototype of the DC-ARQ mechanism is part of J. Morillo-Pozo's Thesis, while its adaptation to the simulator and the performance analysis was carried out by the author of this Thesis, as well as the design, implementation and evaluation of the Carry&forward-ARQ mechanism.

2.2.2 Cooperative Download in Vehicular Environments (51)

O. Trullols-Cruces, M. Fiore, J.M. Barcelo-Ordinas

IEEE Transactions on Mobile Computing (TMC), Issue 99, May 2011

Impact Factor or CORE rank: 2.647, A* Cited by: 16 [1]

Contribution: This paper takes as starting point M. Fiore and J.M. Barcelo paper (60), published in IEEE MASS'09. The author of this thesis designed, implemented and evaluated the oracle algorithm, carried out more extensive simulations, carried out the extended analysis of the results, under the supervision of the other authors. The first version of the simulator we used, was developed by Marco Fiore for (60), all subsequent work, adapting the simulator to the extended scenario and more detailed outputs, as well as the improvements and refinements to the algorithms presented in the paper were done by the author of this thesis.

[1] Google scholar citations April 2014

2.2.3 Other publications (3)

A Cooperative ARQ for Delay-Tolerant Vehicular Networks (47)

J. Morillo-Pozo, J. M. Barcelo-Ordinas, **O. Trullols-Cruces**, J. Garcia-Vidal
IEEE International Workshop on Delay-Tolerant Mobile Networks (DTMN'08), June 2008
Impact Factor or CORE rank: A Cited by: 18[1]

Contribution: My contribution to this work has been the discussion, participation (includes driving one of the cars during the experiments) and results interpretation in the vehicular experiments carried out for the paper.

Evaluation of a Cooperative ARQ Protocol for Delay-Tolerant Vehicular Networks (48)

J. Morillo-Pozo, J. M. Barcelo-Ordinas, **O. Trullols-Cruces**, J. Garcia-Vidal
Lecture Notes in Computer Science (LNCS), Volume 4396, pp 43-61, July 2008
Impact Factor or CORE rank: 0.4 Cited by: 4[1]

Contribution: My contribution to this work has been the discussion, participation (includes driving one of the cars during the experiments) and results interpretation in the vehicular experiments carried out for the paper.

Applying Cooperation for Delay Tolerant Vehicular Networks (49)

J. Morillo-Pozo, J. M. Barcelo-Ordinas, **O. Trullols-Cruces**, J. Garcia-Vidal
4th EuroFGI Workshop on Wireless and Mobility, Barcelona, Spain, September 2008
Impact Factor or CORE rank: - Cited by: 7[1]

Contribution: My contribution to this work has been the discussion, participation (includes driving one of the cars during the experiments) and results interpretation in the vehicular experiments carried out for the paper, as well as the design, implementation and evaluation of the Carry&forward-ARQ mechanism.

2.3 Roadside Infrastructure Deployment for Information Dissemination

2.3.1 Planning Roadside Infrastructure for Information Dissemination in Intelligent Transportation Systems (54)

O. Trullols-Cruces, M. Fiore, C. Casetti, C.F. Chiasserini, J.M. Barcelo-Ordinas, Computer Communications, Elsevier, January 2010

Impact Factor or CORE rank: 0.933 Cited by: 45 [1]

Contribution: This paper is an extension of our previous work published at IEEE WiMob, containing a more detailed evaluation of the deployment algorithm, as well as other results that could not fit the Conference paper format limitations. My contribution to this extension has been the design, implementation and evaluation of the Road Side unit Deployment algorithms, under the supervision and guidance of the other authors.

2.3.2 A Max Coverage Formulation for Information Dissemination in Vehicular Networks (53)

O. Trullols-Cruces, M. Fiore, C. Casetti, C.F. Chiasserini and J.M. Barcelo

IEEE Int. Conf. on Wireless and Mobile Computing, Networking and Communications, Marrakech, Morocco, October 2009

Impact Factor or CORE rank: C Cited by: 4 [1]

Contribution: My contribution to this work has been the problem formulation, design, implementation and evaluation of the Road Side unit Deployment algorithms, under the supervision and guidance of the other authors.

2.3.3 RSU Deployment for Content Dissemination and Downloading in Intelligent Transportation Systems (55)

M. Reineri, C. Casetti, C.F. Chiasserini, M. Fiore, **O. Trullols-Cruces** and J.M. Barcelo

Book Chapter to "Roadside Networks for Vehicular Communications: Architectures, Applications and Test Fields", December 2012

Impact Factor or CORE rank: - Cited by: 1 [1]

Contribution: It's an extension of our Elsevier Computer Communications work, merged with M. Reineri's results. The RSU deployment algorithms were presented in our previous works, some new scenarios were run for this publication, while the previous simulations needed to be rerun to obtain more detailed results needed by M. Reineri, for the fine grained simulations with ns-2 of the smaller scenarios presented.

2.4 Malware in MANETS

2.4.1 Understanding, Modeling and Taming Mobile Malware Epidemics in a Large-scale Vehicular Network (56)

O. Trullols-Cruces, M. Fiore, J.M. Barcelo-Ordinas,

IEEE 14th World of Wireless, Mobile and Multimedia Networks (WoWMoM), Madrid, Spain, 2013

Impact Factor or CORE rank: A Cited by: 0 [1]

Contribution: My contribution to this work has been the problem formulation, design and implementation of the simulator, evaluation of the results and the mathematical malware spread model development under the supervision and guidance of M. Fiore and J.M. Barcelo-Ordinas.

2.5 MANET Power Saving Trade-offs

2.5.1 Power Saving Trade-offs in Delay/Disruptive Tolerant Networks (57)

O. Trullols-Cruces, J. Morillo, J. M. Barcelo and J. Garcia-Vidal,

IEEE 12th World of Wireless, Mobile and Multimedia Networks (WoWMoM), Lucca, Italy, June 2011

Impact Factor or CORE rank: A Cited by: 6 [1]

Contribution: My contribution to this work has been the design, evaluation of the power saving analytical model, as well as the implementation of the simulations to validate the model, under the supervision and guidance of the other authors.

2.6 Analysis of Random Linear Network Coding

2.6.1 Exact Decoding Probability Under Random Linear Network Coding (59)

O. Trullols-Cruces, J.M. Barcelo-Ordinas, M. Fiore,

IEEE Communication Letters, January 2011, ISSN:1089-7798, (online)November 2010

Impact Factor or CORE rank: 1.140 Cited by: 30 [1]

Contribution: My contribution to this work has been the definition, resolution and validation using simulations of the decoding problem, under the guidance and supervision of J.M. Barcelo-Ordinas and M. Fiore.

2.7 Other publications

2.7.1 Generation and Analysis of a Large-scale Urban Vehicular Mobility Dataset (36)

S. Uppoor, **O. Trullols-Cruces**, M. Fiore, J.M. Barcelo-Ordinas

IEEE Transactions on Mobile Computing (TMC), february 2013 (online, preprint)

Impact Factor or CORE rank: 2.647, A* Cited by: 8 [1]

Contribution: My contribution to this work is focused primarily in the second part of the paper, in the analysis of the impact of show the potential impact that such mobility datasets have on the network protocol performance analysis.

2.7.2 VANET Mobility Modeling Challenged by Feedback Loops (Invited Paper) (37)

H. Meyer, **O. Trullols-Cruces**, A. Hess, K.A. Hummel, J.M. Barcelo, C.E. Casetti and G. Karlsson,

10th IEEE/IFIP Annual Mediterranean Ad Hoc Networking Workshop (IEEE/IFIP Med-Hoc-Net), June 2011

Impact Factor or CORE rank: - Cited by: 5 [1]

Contribution: My contribution to this work has been the participation in the project meetings and discussions, as well as using my simulator on different mobility traces (generated by the other authors) to obtain the results presented in the paper.

3

Cooperative Download in Vehicular Environments

3.1 Introduction

Vehicles traveling within cities and along highways are commonly regarded as most probable candidates for a complete integration into mobile networks of the next generation. Vehicle-to-infrastructure and vehicle-to-vehicle communication could indeed foster a number of new applications of notable interest and critical importance, ranging from danger warning to traffic congestion avoidance. It is however easy to foresee that the availability of on-board communication capabilities will also determine a significant increase in the number of mobile users regularly employing business and infotainment applications during their displacements. As a matter of fact, equipping vehicles with WiMAX/LTE and/or Wi-Fi capabilities would represent a clear invitation for passengers on cars or buses to behave exactly as home-based network users. The phenomenon would thus affect not only lightweight services such as web browsing or e-mailing, but also resource-intensive ones such as streaming or file sharing.

Although this could not represent a problem for relatively lightweight services such as web browsing or e-mailing, resource-intensive tasks such as video streaming or file sharing will instead risk to overload the wireless communication infrastructure. This could result in much worse breakdowns than those already faced today by cellular networks in front of the growing number of high-end mobile users.

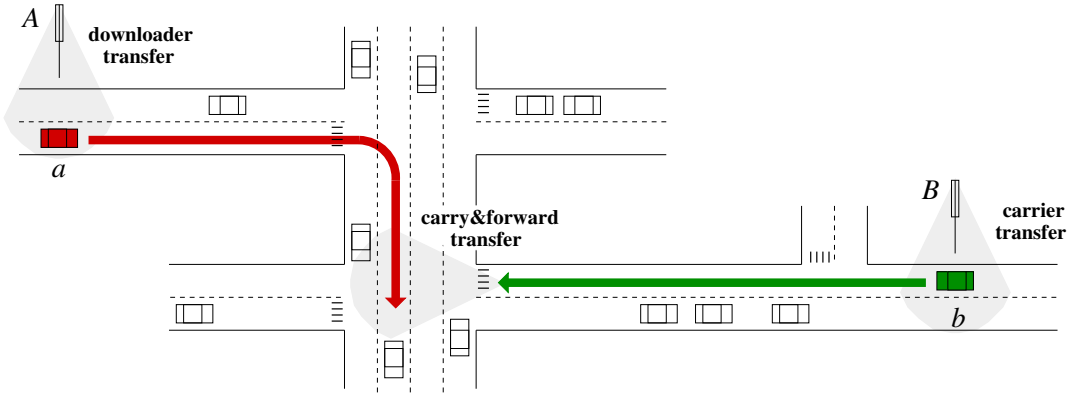


Figure 3.1: Vehicle a downloads part of some content from AP A. The idle AP B delegates another portion of the same content to a vehicle b. When b encounters a, vehicle-to-vehicle communication is employed to transfer to a the data carried by b

In order to support these demanding operations and thus favor the network scalability, a valuable aid to the traditional user-to-infrastructure communication paradigm could come from interactions among mobile users. Within such context, the fast movement dynamics that characterize vehicular environments make fully ad-hoc approaches, that try to build a connected network over moving cars, impractical. Instead, opportunistic vehicle-to-vehicle communication appears as a more viable complement to infrastructure-based connectivity.

In this chapter, we focus on one of the latter tasks, namely the download of large-sized files from the Internet. More precisely, we consider a urban scenario, where users aboard cars can exploit roadside Access Points (APs) to access the servers that host the desired contents. We consider that the coverage provided by the roadside APs is intermittent: this is often the case, since, in presence of large urban, suburban and rural areas, a pervasive deployment of APs dedicated to vehicular access is often impractical, for economic and technical reasons. We also assume that not all on-board users download large files all the time: indeed, one can expect a behavior similar to that observed in wired networks, where the portion of queries for large contents is small (61). As a result, only a minor percentage of APs is simultaneously involved in direct data transfers to downloader cars in their respective coverage area, and the majority of APs is instead idle.

Within such a context, we study how opportunistic vehicle-to-vehicle communication can complement the infrastructure-based connectivity, so to speed up the down-

load process. More precisely, we exploit the APs inactivity periods to transmit, to cars within range of idle APs, pieces of the data being currently downloaded by other vehicles. Cars that obtain information chunks this way can then transport the data in a *carry&forward* fashion (9), and deliver it to the destination vehicle, exploiting opportunistic contacts with it. For the sake of clarity, a simple example of this approach is provided in Fig. 3.1. We remark that the concept of cooperative download in vehicular networks has been already proposed for highway environments: however, unlike what happens over unidimensional highways, urban/suburban road topologies present multiple route choices that make it hard to predict if vehicles will meet; moreover, the presence of traffic lights, stop and yield signs renders cars contact timings very variable. These key aspects make highway-tailored solutions impracticable in complex non-linear road scenarios, for which we are, to the best of our knowledge, the first to identify challenges and propose solutions.

After a discussion of the literature, in Sec. 3.2, we outline the major challenges of vehicular cooperative download in urban environments and devise original solutions to them, in Sec. 3.3. In Sec. 3.4 we present the scenarios considered for our performance evaluation, whose results are then discussed in Sec. 3.5. Conclusions are drawn in Sec. 3.6.

3.2 Related work

The cooperative download of contents from users aboard vehicles has been first studied in (62), that introduced SPAWN, a protocol for the retrieval and sharing of contents vehicular environments. SPAWN is designed for unidirectional traffic over a highway, and is built on the assumption that all on-road vehicles are active downloaders of a same content. Instead, we target urban environments where users may be interested in different contents. Two major aspects distinguish our work from SPAWN. First, SPAWN is designed for unidirectional traffic over a highway, and thus does not address any of the challenges that we outline in this chapter and that are characteristic of more complex road scenarios. Second, SPAWN is inspired by peer-to-peer networking, and thus is built on the assumption that all on-road vehicles are active downloaders of a same content; instead, we consider the more general case of users interested in different files. Similar considerations hold for the works in (63) and (50).

In (64), the highway scenario is replaced by a circular bus route within a campus, which however implies again easily predictable vehicular contacts: indeed, the focus of the work is on the prefetching and multi-hop transfer of data at each individual AP, while carry&forward communications are not taken into consideration. Conversely, (65) and (66) examine urban environments. In (65), the authors study the upload of small-sized contents from vehicles to roadside gateways, rather than the large downloads we target. The work in (66) considers instead data transfers to vehicular users in grid-like road topologies, but the focus is on the problem of optimizing direct communications between cars and infrastructure, without taking into account cooperation among mobile users. Recently, the performance bounds of vehicular cooperative download in urban scenarios have been studied in (67): there, however, the authors assume perfect knowledge of the car traffic and outline a centralized optimal solution, rather than the distributed practical techniques we envisage in this work.

As far as opportunistic data exchanges are concerned, the potential of such a networking paradigm in vehicular environments was first shown in (68), further explored in (8, 69), and exploited in (70, 71) among the others. However, most of these works focus on routing delay-tolerant information in vehicular networks, while none copes with the problem of cooperative download. Also, techniques for Medium Access Control (72) and network coding (73) that have been proposed for cooperative vehicular download are orthogonal to the problems we address, and could complement the solutions outlined in this chapter.

Finally, since we study the impact of the infrastructure deployment on the cooperative download, our work also relates to the topic of AP placement in vehicular networks. In (74), the authors studied the impact of random AP deployments on data routing in urban road topologies: we will prove that such an approach is inefficient when targeting cooperative download. More complex solutions for the deployment of APs over road topologies have been proposed in (75), to favor delay-tolerant data exchange among vehicles, and in (76), for information dissemination purposes. However, the diverse goals in these works lead to in different approaches and results with respect to ours. More recently, the problem of AP placement to provide Internet access to vehicles has been addressed in (77, 78) and (54). In all these works, however, the aim is to maximize vehicle-to-infrastructure coverage or contacts, and no cooperation among cars is considered.

3.3 Cooperative download

Let us first point out which are the major challenges in the realization of a vehicular cooperative download system within complex urban road environments. With reference to the transfer model proposed in Sec. 3.1, we identify two main problems:

- *the selection of the carrier(s)*: contacts between cars in urban/suburban environments are not easily predictable. Idle APs cannot randomly or inaccurately select vehicles to carry data chunks, or the latter risks to be never delivered to their destinations. Choosing the right carrier(s) for the right downloader vehicle is a key issue in the scenarios we target;
- *the scheduling of the data chunks*: determining which parts of the content should be assigned to one or multiple carriers, and choosing in particular the level of redundancy in this assignment, plays a major role in reducing the probability that destination vehicles never receive portions of their files.

In the following, we first discuss the selection of carriers at the APs, proposing to leverage historical information on large-scale traffic flows to drive data transfer decisions. Then, we outline several solutions to the chunk scheduling problem, that are characterized by different levels of redundancy.

3.3.1 Carriers selection

The first problem we address is that of the selection of data chunk carriers at APs that are idle, i.e., that are currently not transferring data directly to vehicular downloaders. As previously discussed, these APs can opt to employ their spare airtime to delegate, to mobile users within range, portions of files being downloaded. Taking such a decision means to answer to two questions: (i) *which, among the vehicles in range of an idle AP, should be picked as carriers, if any?* and (ii) *which of the downloaders should these carriers transport data for?* As we already underlined, carriers selection is performed only if there are no vehicles interested in a direct download in range of the AP, as, in such case, priority is given to them.

The key to the answers is to know in advance whether (and possibly when) one or more cars currently within coverage of an AP will meet a specific downloader vehicle, so

to perform the selection that maximizes the download rate. Also, by choosing carriers depending on their future contacts, the destination of the data becomes constrained to the elected carriers, among vehicles currently engrossed in downloading content and the second question above is inherently solved along with the first one. However, assuming that the roadside infrastructure has perfect knowledge of the future route of each user is unrealistic, other than raising privacy issues. At the same time, the movement of individual vehicles over urban road topologies cannot be easily predicted as in unidimensional highways. We then adopt a probabilistic approach, by leveraging the fact that large-scale urban vehicular flows tend to follow common movement patterns (79, 80, 81). More precisely, the solution we propose leverages *contacts maps*, that are built by exploiting historical data on contacts between car flows, and then used to estimate the meeting probability between downloaders and candidate data carriers.

In the following, we first discuss the structure and construction of contacts maps. Then, we present different carriers selection algorithms, detailing how they exploit such maps.

Contacts map

We denote as p^k_{Aa} the k-th *production phase* of vehicle a with respect to AP A, i.e., the k-th of the disjoint time intervals during which vehicle a can steadily download data from A (6). From a specific AP perspective, we tag production phases as *local* if they involve that particular AP: as an example, p^h_{Bb} is a local production phase for AP B, $\forall\, b, h$. On the other hand, we label as f^m_{ab} the m-th *forward phase* of vehicle b with respect to vehicle a, i.e., the m-th of the disjoint time intervals during which vehicle b can steadily forward data to vehicle a. Note that production and forward phases do not necessarily correspond to actual data transfers, but just to contacts which could be exploited for data transfers. We also use $t(\cdot)$ to indicate the time at which a production or forward phase starts, and $\Delta t(\cdot)$ to tag its duration. For production phases only, $\alpha(\cdot)$ denotes the general direction of movement[1] of the vehicle at the beginning of the production phase, and $v(\cdot)$ its speed at that same time. The notation is summarized in Fig. 3.2.

[1]The general direction is obtained as the angle of movement between the location where the vehicle started its trip, and its current location. This represents a more reliable information than the instantaneous direction, and it is not harder to obtain from a GPS receiver than the latter.

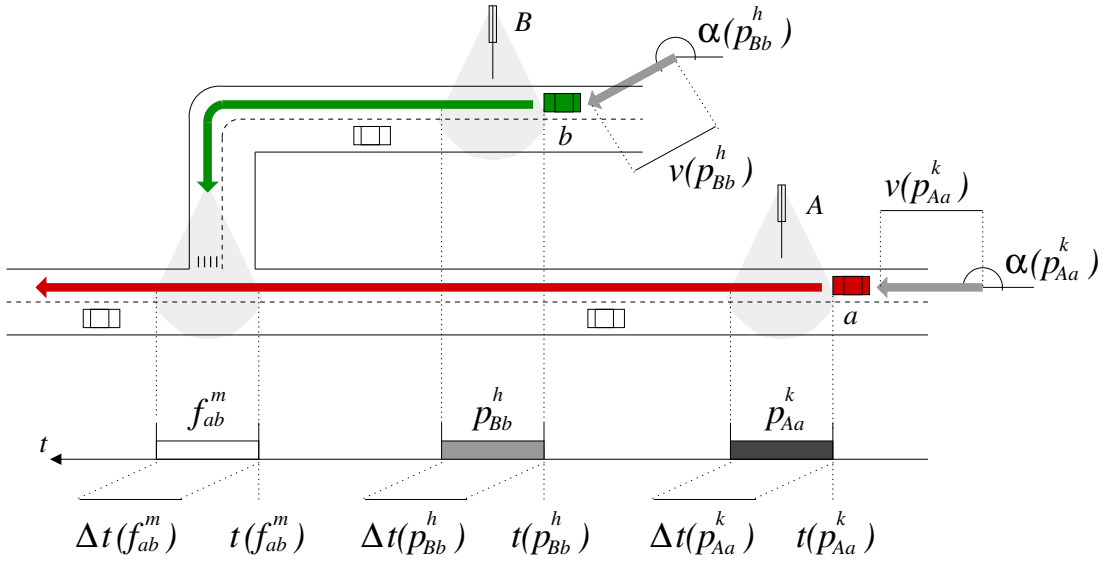


Figure 3.2: Notation for contacts map structure and construction

Structure. A contacts map is a data structure that provides an AP with information on the probability of contact between a vehicle involved in a local production phase and another vehicle. With reference to the example in Fig. 3.2, the contacts map at AP B allows B to know the probability of contact between the local vehicle b and the generic vehicle a. In particular, AP B knows that b started a local production phase p^h_{Bb} at time $t(p^h_{Bb})$, while moving with direction $\alpha(p^h_{Bb})$ and speed $v(p^h_{Bb})$; also, let us assume B has been informed that a started a production phase p^k_{Aa} with AP A at time $t(p^k_{Aa})$, while moving with direction $\alpha(p^k_{Aa})$ and speed $v(p^k_{Aa})$. Then, the contacts map at B allows to associate the couple of production phases (p^h_{Bb}, p^k_{Aa}) to historical data on the encounters between vehicles that have previously generated production phases at the two APs B and A with timings and mobility similar to those of b and a. We stress that such historical data refers to any two vehicles with movement patterns akin to those of b and a, and not to b and a only: thus, the data concerns vehicular flows rather than individual couples of cars.

More formally, a contacts map is a set of one-to-one associations between *keys*, that encode the significant characteristics of two production phases, and *values*, that store the contacts properties for all couples of production phases that share such characteristics. The *key* for two generic production phases p^h_{Bb} and p^k_{Aa} is a vector $\Bbbk(p^h_{Bb}, p^k_{Aa})$ of the form

$$\left[A, \lfloor \frac{t(p^h_{Bb}) - t(p^k_{Aa})}{\delta t} \rfloor, \lfloor \frac{\alpha(p^h_{Bb})}{\delta \alpha} \rfloor, \lfloor \frac{\alpha(p^k_{Aa})}{\delta \alpha} \rfloor, \lfloor \frac{v(p^h_{Bb}) - v(p^k_{Aa})}{\delta v} \rfloor \right],$$

where $\delta\alpha$, δt and δv are the units (in degrees, seconds, and meters/second, respectively) used to discretize angles, times and speeds. A couple of production phases is thus characterized by the identity of the AP involved in the second production phase, the time elapsed between the start of the two production phases, the direction of the two vehicles at the beginning of the respective production phases, and the difference between their speeds at that same time. We remark that the identity of the other AP is not necessary: as detailed next, the first production phase is always a local one. A *value* is instead a vector of four fields $\{n_{opps}, n_{cons}, t_{del}, t_{dur}\}$:

1. n_{opps}, the number of contact opportunities, i.e., the number of times that the AP observed a couple of production phases with characteristics as from the associated key;

2. n_{cons}, the number of actual contacts, i.e., the number of times that vehicles from the aforementioned couples of production phases actually generated a forward phase;

3. t_{del}, the average time elapsed between the start of the last production phase and the start of the forward phase, if any of the latter has ever occurred;

4. t_{dur}, the average duration of the forward phase, if any has ever occurred.

It is to be noticed that each AP builds its own contacts map, in which it stores only values associated to keys where the first production phase, as already said, is a local one. As an example, an AP B will only store values for keys of the type $\mathbb{k}(p^h_{Bb}, p^k_{Aa})$, $\forall h, b, k, A, a$. The rationale is that local production phases represent the vehicle-to-infrastructure contacts that an AP can exploit for carriers selection, and are thus the only an AP is interested to record data for.

Construction. The steps for the construction of the contacts map at an AP are best described by means of an example, so we consider once more the situation depicted in Fig. 3.2. When the production phase p^k_{Aa} starts, the AP A logs the time $t(p^k_{Aa})$, the relative vehicle identifier a, its general direction $\alpha(p^k_{Aa})$, and its current speed $v(p^k_{Aa})$.

This information is shared, via the wired backbone, with other APs in the same area, and updated, when the production phase ends, with the information on the duration $\Delta t(p^k_{Aa})$. This way, when the production phase p^k_{Aa} terminates, AP B has memory of the event, including all related details. Similarly, at the beginning of p^h_{Bb}, B records and shares with other APs identical information on the production phase, which is then updated at the end of p^h_{Bb}.

Regarding the exchange of data among the APs, we emphasize that a generic AP does not need to be informed about the vehicle-to-infrastructure contacts occurring at all other APs in the network. Indeed, if two APs are too far apart, they can avoid sharing production phase data, as the excessive distance makes contacts too hard to predict and leads to unbounded carry&forward transfer delays. Thus, in order to limit the traffic on the wired backbone and guarantee system scalability, the reciprocal exchange of information about production phases can be constrained to APs within limited geographical distance[1].

Proceeding in our example, at the end of p^h_{Bb}, AP B, as every other AP does at the end of its own local production phases, checks whether p^h_{Bb} can be considered as an opportunity for cooperative download with respect to other production phases it is aware of, i.e., if another production phase is *compatible* with p^h_{Bb}. We will discuss production phases compatibility later in this section; for the moment, let us assume that p^k_{Aa} is compatible with p^h_{Bb}. Then, B looks in its local contacts map for the value associated to key $\mathbb{k}(p^h_{Bb}, p^k_{Aa})$. If an entry is not found, it is created; in both cases, the n_{opps} field in the entry is incremented.

Let us now assume that, later on, vehicle b meets vehicle a, generating the forward phases f^m_{ab} and f^m_{ba}. We focus on the first one, as it is that of interest in our example. Both vehicles record the forward phase start time $t(f^m_{ab})$, as well as the other vehicle identifier. Upon loss of contact, a and b also log the forward phase duration $\Delta t(f^m_{ab})$. These same cars upload these information, together with similar data on all other forward phases they have experienced, to the next AP they encounter, which will again share them with the APs in the area.

[1] A thorough study of the management of control messages over the backbone of the infrastructured network is out of the scope of this chapter. In our tests we imposed a maximum distance of 10 km among data-sharing APs, so to bound the delay between production and forward phases at approximately 15 minutes, given an average vehicular speed of 20 km/h in urban areas.

When AP B is notified of the forward phase f_{ab}^m, it tries to understand if f_{ab}^m can be related to any of the opportunities it has previously recorded. Thus, B scans its database for local production phases compatible with f_{ab}^m; once more, we will discuss the compatibility between production and forward phases next. Assuming that p_{Bb}^h satisfies the compatibility constraint, B then looks for production phases of vehicle a, with any AP, that are compatible with p_{Bb}^h. B finds again p_{Aa}^k, and thus finally relates f_{ab}^m to the couple of production phases (p_{Bb}^h, p_{Aa}^k). At this point, B retrieves the value associated to the key $\Bbbk(p_{Bb}^h, p_{Aa}^k)$, and updates the n_{cons}, t_{del} and t_{dur} fields. The first is incremented by one, to record that the opportunity previously stored actually generated a forward phase. The second and the third elements are updated using samples $t(f_{ab}^m) - t(p_{Bb}^h)$ and $\Delta t(f_{ab}^m)$, respectively. The way these last updates are performed depends on the desired level of detail on the vehicle-to-vehicle contact: in our case, we opted for keeping track of the mean of the samples.

Phases compatibility. Phases compatibility rules determine when two production phases generate an opportunity for cooperative download, as well as when a forward phase can represent a contact for a local production phase. These rules formally relate phases in a way that avoids inconsistencies in the resulting contacts maps, an event otherwise common, especially when considering that phases often overlap in time and/or refer to a same AP (i.e., it can be that A and B in all previous discussions are indeed the same AP). We first introduce the set of rules for the compatibility of a local production phase with respect to a forward phase. A local production phase p_{Bb}^h is said to be compatible with a forward phase f_{ab}^m if the following conditions are verified:

1) the forward phase has ended after the end of the local production phase, or

$$t(f_{ab}^m) + \Delta t(f_{ab}^m) > t(p_{Bb}^h) + \Delta t(p_{Bb}^h),$$

as b must receive the data from B before it can forward them to a. Note that the position of the subscripts already implies that, for the phases to be compatible, the same vehicle b must realize the production phase with B and be the potential carrier in the forward phase;

2) the forward phase is the first involving a and b and satisfying rule 1) above to have terminated after the end of the local production phase, or

$$\nexists n \mid t(f_{ab}^n) + \Delta t(f_{ab}^n) > t(p_{Bb}^h) + \Delta t(p_{Bb}^h),$$
$$t(f_{ab}^n) < t(f_{ab}^m),$$

which guarantees that at most one forward phase is associated to each production phase;

3) the local production phase at B is the last, involving b and satisfying rule 1) above, that started before the forward phase end, or

$$\nexists n \mid t(f_{ab}^m) + \Delta t(f_{ab}^m) > t(p_{Bb}^n) + \Delta t(p_{Bb}^n),$$
$$t(p_{Bb}^n) > t(p_{Bb}^h),$$

which guarantees that at most one production phase is associated to each forward phase.

Then, we introduce the rules that define the compatibility between two production phases. A production phase p_{Aa}^k is said to be compatible with a local production phase p_{Bb}^h if the following conditions are verified:

4) the first production phase has ended before the end of the local production phase, or

$$t(p_{Bb}^h) + \Delta t(p_{Bb}^h) > t(p_{Aa}^k) + \Delta t(p_{Aa}^k),$$

which accounts for the fact that an AP can only destine carry&forward data to production phases it is aware of;

5) the first production phase has ended at most a time T before the end of the local production phase, or

$$t(p_{Bb}^h) + \Delta t(p_{Bb}^h) - t(p_{Aa}^k) - \Delta t(p_{Aa}^k) \leq T,$$

that avoids considering obsolete production phases;

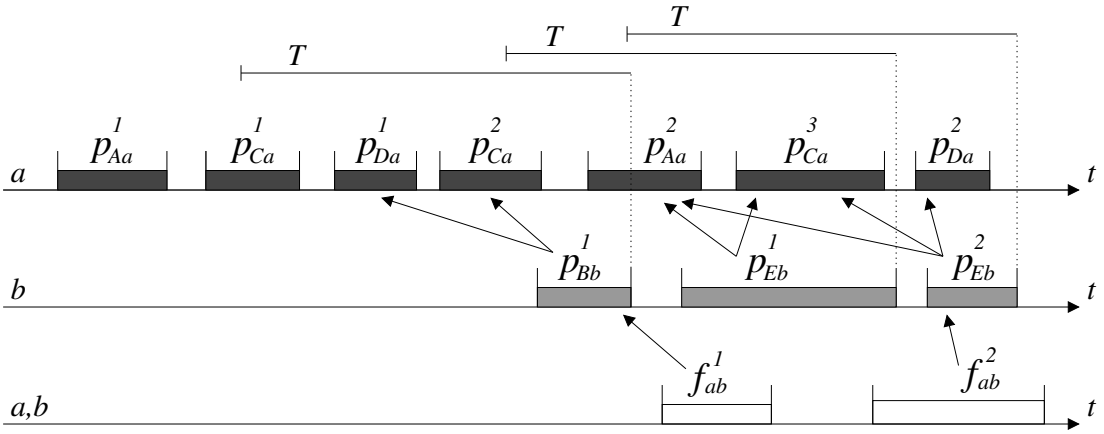


Figure 3.3: Example of phases compatibility. Arrows show production phases of car a that are compatible with local production phases of car b, as well as which of the latter are compatible with forward phases of b to a

6) the first production phase is the last, involving a and A as well as satisfying rules 1) and 2) above, to have ended before the end of the local production phase

$$\nexists n \mid 0 < t(p^h_{Bb}) + \Delta t(p^h_{Bb}) - t(p^n_{Aa}) - \Delta t(p^n_{Aa}) \leq T,$$
$$t(p^n_{Aa}) > t(p^k_{Aa}),$$

which guarantees that at most one production phase involving same vehicle/AP couple is associated to each local production phase.

Fig. 3.3 provides some examples of phases compatibility as from the rules listed above. There, the first timeline depicts the time intervals during which a vehicle a experiences production phases with APs A, C and D, while the second timeline reports the sequence of production phases of car b with APs B and E. Finally, the last timeline shows when forward phases of b to a occur.

Arrows from the second to the first timeline indicate which production phases of a are compatible with those of b. As an example, p^1_{Aa} is not compatible with p^1_{Bb} because it occurred too early in time (i.e., it ended more than a time T before the end of p^1_{Bb}, see rule 5 above), while p^2_{Aa} is not compatible with p^1_{Bb} because it is not yet concluded when p^1_{Bb} ends (rule 4). Similarly, p^1_{Ca} is not compatible with p^1_{Bb} since a more recent production phase between a and C, i.e., p^2_{Ca}, is compatible with p^1_{Bb} (rule 6). Conversely, p^2_{Ca} and p^1_{Da} satisfy all the compatibility conditions, and are thus compatible with p^1_{Bb}.

Forward phase compatibilities are shown as arrows from the third to the second timeline. We can notice that f^2_{ab} is compatible with p^2_{Eb} but not with p^1_{Eb}, since p^1_{Eb} is not the last production phase involving b and E to have ended before the end of f^2_{ab} (rule 3 above), while p^2_{Eb} is. Moreover, f^2_{ab} is not compatible with p^1_{Bb}, because another forward phase of b to a, i.e., f^1_{ab}, already concluded after the end of p^1_{Bb} (rule 2).

Carriers selection algorithms

Contacts maps can be exploited by APs to select local cars as data carriers in the cooperative download process, by retrieving their contact probability estimates with respect to downloader vehicles. Firstly, it is necessary that APs know which cars in their surroundings are interested in some content. Thus, every time a downloader vehicle starts a production phase, the fact that it is requesting data, as well as the nature of the desired content, is attached to the usual information on the production phase that the local AP shares with other APs. This way, each AP can track downloaders through their production phases history.

Thanks to such knowledge, an AP that has active local production phases can compute the *delivery potential* $\mathbb{p}_a$ resulting from electing one (or some, or all) of the local vehicles as carrier(s) for data destined to a specific downloader vehicle a. The delivery potential is obtained as the sum of the individual contact probabilities $\mathbb{p}_b$, derived from assigning data for the downloader a to each elected local carrier b[1]. The process is repeated for each known downloader car, and, in the end, the downloader vehicle associated with the highest delivery potential $\mathbb{p}$ is chosen as the target of a carry&forward transfer through local carriers that contributed to $\mathbb{p}$. Note that $\mathbb{p}$ is a potential and not a probability: indeed, $\mathbb{p}$ can be higher than one to counter uncertainties in probability estimates.

The framework for carriers selection run at a generic AP B is shown as pseudocode in Fig. 3.4. There, priority is always given to direct data transfers to downloader cars, and fairness among them is provided by always picking the vehicle that is the closest to the AP. The parameter $\mathbb{P}_{min}$ controls the minimum delivery potential required to attempt cooperative download through local carriers. The value of such parameter (line 01 in Fig. 3.4), together with the way the delivery potential $\mathbb{p}^k_{Aa}$ associated to the

[1]Thanks to the broadcast nature of the wireless channel, a single transmission is sufficient to transfer the same data to all elected local carriers.

```
01  set 𝕡 equal to ℙ_min
02  for each downloader vehicle a do
03    if a is in range of B do
04      if a is closer to B than previous direct downloaders do
05        select a as destination for direct transfer
06        select no vehicles as carriers for carry&forward transfer
07        set 𝕡 equal to ∞
08      done
09    else
10      for each production phase p^k_Aa of a do
11        until all on-going local production phases are not processed do
12          update delivery potential 𝕡^k_Aa
13          update carriers list c̄^k_Aa
14        done
15        if 𝕡^k_Aa is the highest potential computed for a do
16          set 𝕡_a equal to 𝕡^k_Aa
17          set c̄_a equal to c̄^k_Aa
18        done
19      done
20      if 𝕡_a is strictly higher than 𝕡 do
21        select a as destination for carry&forward transfer
22        select vehicles in c̄_a as carriers for carry&forward transfer
23        set 𝕡 equal to 𝕡_a
24      done
25    done
26  done
```

Figure 3.4: Pseudocode for carriers selection at AP B

downloader production phase p^k_{Aa} and its relative carriers list $\bar{\mathbb{c}}^k_{Aa}$ are updated (lines 12 and 13 in Fig. 3.4), distinguish the following carriers selection algorithms.

The **Blind** carriers selection algorithm aims at fully exploiting the airtime available at APs, by delivering data to all available local carriers whenever possible. This algorithm does not make use of the contacts map, but randomly chooses a downloader car as the destination of the data: we thus employ it as a benchmark for the other schemes. The pseudocode for potential and carriers list updating is outlined in Fig. 3.5, while

```
01  get next on-going local production phase p^h_Bb
02  set 𝕡_b equal to a random value ∈ (0, 1]
03  add 𝕡_b to delivery potential 𝕡^k_Aa
04  add b to carriers list c̄^k_Aa
05  mark local production phase p^h_Bb as processed
```

Figure 3.5: Blind pseudocode for $\mathbb{p}^k_{Aa}$, $\bar{\mathbb{c}}^k_{Aa}$ update

```
01 get next on-going local production phase p^h_Bb
02 get key k(p^h_Bb, p^k_Aa)
03 if a contacts map entry for such key exists do
04   get relative value {n_opps, n_cons, t_del, t_dur}
05   set p_b equal to n_cons / n_opps
06   add p_b to delivery potential p^k_Aa
07   add b to carriers list c̄^k_Aa
08 done
09 mark local production phase p^h_Bb as processed
```

Figure 3.6: p-Driven pseudocode for $\mathbb{p}^k_{Aa}$, $\bar{\mathbb{c}}^k_{Aa}$ update

```
01 get next on-going local production phase p^h_Bb
02 get key k(p^h_Bb, p^k_Aa)
03 if a contacts map entry for such key exists do
04   get relative value {n_opps, n_cons, t_del, t_dur}
05   set p_b equal to n_cons / n_opps
06   if p_b is equal to or greater than P_ind do
07     add p_b to delivery potential p^k_Aa
08     add b to carriers list c̄^k_Aa
09   done
10 done
11 mark local production phase p^h_Bb as processed
```

Figure 3.7: p-Constrained pseudocode for $\mathbb{p}^k_{Aa}$, $\bar{\mathbb{c}}^k_{Aa}$ update

$\mathbb{P}_{min}$ is set to 0, so that cooperative download is always attempted when at least one local carrier is present.

The **p-Driven** carriers selection algorithm is a probability-driven version of the Blind algorithm above. It again tries to exploit as much as possible the APs wireless resources, but this time cooperative download destinations are selected according to the delivery potential obtained from the contacts map.

As a matter of fact, carry&forward data is consigned by each AP to all available local vehicles, and destined to the downloader vehicle which maximizes the sum of its contact probabilities with all the local carriers, as detailed in the pseudocode of Fig. 3.6. We stress that non-compatible production phases generate keys that are not present in the contacts map, and are thus not considered for cooperative download. As the p-Driven algorithm is designed to exploit carry&forward whenever there is a minimal chance of delivery, $\mathbb{P}_{min}$ is set to 0: this allows cooperative download even in presence of very small delivery potentials. Exploiting contacts maps, the p-Driven scheme is however expected to be more precise than the Blind one in the selection of carriers.

The **p-Constrained** carriers selection algorithm builds on top of the p-Driven scheme, adding constraints on probabilities, as from the pseudocode in Fig. 3.7. In particular, local vehicles with individual contact probability $\mathbb{p}_b$ lower than $\mathbb{P}_{ind} > 0$ are not considered for data carrying, and $\mathbb{P}_{min}$ is set to a value higher than 0, so that downloader vehicles with delivery potential $\mathbb{p}_a$ lower than $\mathbb{P}_{min}$ are discarded. Thanks to the lower bounds on individual probability and delivery potential, the p-Constrained algorithm is expected to further increase the delivery precision and reduce the load at APs with respect to the p-Driven scheme. However, quality could come at cost of quantity, as the thresholds may hinder potentially successful cooperation among vehicles.

The **(p,t)-Constrained** carriers selection algorithm adds time constraints to the probability bounds of the p-Constrained scheme. It introduces a distributed database $\bar{\mathbb{t}}_a$, maintained for each active downloader vehicle a by APs in a same area, controlling what portions of a's airtime are assigned to which specific carriers[1]. As shown in the pseudocode in Fig. 3.8, the (p,t)-Constrained algorithm processes local vehicles b in decreasing order of contact probability with the downloader car a, skipping those with probability lower than $\mathbb{P}_{ind}$ (lines 02 to 16 in Fig. 3.8). Every time the unprocessed local vehicle with maximum contact probability $\mathbb{p}_{max}$ is processed, the algorithm exploits information on the average time to contact (t_{del}) and contact duration (t_{dur}) to predict the time interval during which the local vehicle will meet the downloader car a. Then, it discretizes time with step $\mathbb{T}$, and tries to fit the estimated contact probability $\mathbb{p}_{max}$ in one of the time steps within the aforementioned time interval (lines 20 to 30 in Fig. 3.8).

The process is terminated when either the required delivery potential $\mathbb{P}_{max}$ has been reached (lines 31 to 33 in Fig. 3.8), or no more local vehicles are available (lines 17 to 19 in Fig. 3.8). In the second case, the delivery potential constraint is not fulfilled, as $\mathbb{P}_{min}$ is higher than zero, and thus no carriers can be selected for the current production phase p^k_{Aa} (see line 20 in Fig. 3.4). The (p,t)-Constrained algorithm therefore employs information about contact times to improve the delivery precision, reducing the data carriers involved in the cooperative download.

[1]We recognize that maintaining such database can pose synchronization and consistency issues, whose management is out of the scope of this thesis. We however note that we do not require frequent updates or high accuracy in $\bar{\mathbb{t}}_a$, since the update periodicity is in the order of seconds and errors in the database are overshadowed by inaccuracy in the contact estimation.

set $\mathbb{p}_{max}$ equal to $\mathbb{P}_{ind}$
for each on-going local production phase p^h_{Bb} **do**
  if p^h_{Bb} is marked as processed **do**
    continue
  done
  get key $\mathbb{k}(p^h_{Bb}, p^k_{Aa})$
  if a contacts map entry for such key exists **do**
    get relative value $\{n_{opps}, n_{cons}, t_{del}, t_{dur}\}$
    set $\mathbb{p}_b$ equal to $\frac{n_{cons}}{n_{opps}}$
    if $\mathbb{p}_b$ is equal to or greater than $\mathbb{p}_{max}$ **do**
      set p equal to production phase p^h_{Bb}
      set v equal to production phase p^h_{Bb} vehicle b
      set $\mathbb{p}_{max}$ equal to $\mathbb{p}_b$
    done
  done
done
if $\mathbb{p}_{max}$ is equal to $\mathbb{P}_{ind}$ **do**
  mark all unmarked local production phases as processed
else
  get key $\mathbb{k}(p, p^k_{Aa})$
  get relative value $\{n_{opps}, n_{cons}, t_{del}, t_{dur}\}$
  for each time step $t \in \left[\lfloor \frac{t(p)+t_{del}}{\mathbb{T}} \rfloor, \lfloor \frac{t(p)+t_{del}+t_{dur}}{\mathbb{T}} \rfloor\right]$ **do**
    if $\mathbb{p}_{max}$ is lower than or equal to $\mathbb{P}_{min} - \bar{\mathbb{t}}_a(t)$ **do**
      add $\mathbb{p}_{max}$ to delivery potential $\mathbb{p}^k_{Aa}$
      add v to carriers list $\bar{\mathbb{c}}^k_{Aa}$
      set $\bar{\mathbb{t}}_a(t)$ equal to $min\{\bar{\mathbb{t}}_a(t) + \mathbb{p}_{max}, \mathbb{P}_{min}\}$
      continue
    done
  done
  mark local production phase p as processed
  if $\mathbb{p}^k_{Aa}$ is equal to or greater than $\mathbb{P}_{min}$ **do**
    mark all unmarked local production phases as processed
  done
done

Figure 3.8: (p,t)-Constrained pseudocode for $\mathbb{p}^k_{Aa}$, $\bar{\mathbb{c}}^k_{Aa}$ update

3.3.2 Chunk scheduling

Upon selection of a destination for the carry&forward transfer, jointly with the associated local carriers, an AP must decide on which portion of the data the downloader is interested in is to be transferred to the carriers. To that end, we assume that each content is divided into *chunks*, i.e., small portions of data that can be transferred as a single block from the AP to the carriers, and then from the latter to the destination. Since a same chunk can be transferred by one or multiple APs to one or more carriers, the chunk scheduling problem yields a tradeoff between the reliability (i.e., the probability that a downloader will receive at least one copy of a chunk) and the redundancy (i.e., how many copies of a same chunk are carried around the road topology) of the data transfer. Next, we introduce three chunk scheduling schemes that embody growing levels of redundancy, and that are thus intended to provide increasing communication reliability.

The **Global** chunk scheduling assumes that APs maintain per-vehicle distributed chunk databases, similar to the time databases $\bar{\mathfrak{t}}_a$ introduced before[1]. These databases store information on which chunks have already been scheduled for either direct or carry&forward delivery to each downloader.

The Global scheme, whose flow diagram is depicted in Fig. 3.9, completely distributes the chunk scheduling among APs, since it forces an AP to pick a new, unscheduled chunk every time it performs a direct or carry&forward transfer. In other words, each chunk is scheduled for transfer just once in the entire network. We stress that, even then, multiple carriers can be given the same chunk, as carriers selection algorithms can (and usually do) identify more than one vehicle for a single carry&forward transfer.

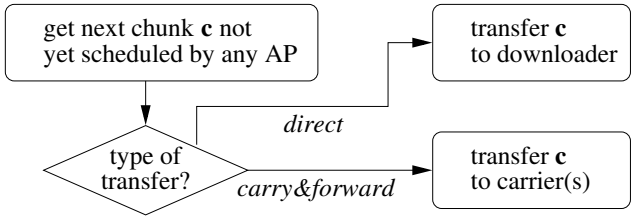


Figure 3.9: Flow diagram of the Global chunk scheduling algorithms

[1]The same observations on database maintenance apply here as well.

The **Hybrid** chunk scheduling, in Fig. 3.10, allows overlapping between carry&forward transfers scheduled by different APs. Indeed, in case of a data transfer to carriers, an AP picks the first chunk that *it* has not yet scheduled, ignoring the carry&forward scheduling at the other APs. Conversely, non-overlapping scheduling is still enforced for direct chunk transfers: every time it has to deliver some portion of a content to a downloader, an AP always selects a new chunk, not yet scheduled by any other AP in the region. The Hybrid scheme is thus implicitly more redundant that the Global one, as different APs independently delegate carriers for a same data chunk. Also, note that the Hybrid scheme does not need the aforementioned per-vehicle chunk databases. As a matter of fact, it commends that overlapping is avoided only for direct transfers, that however occur during contacts between APs and the downloader vehicle: as a consequence, the chunk scheduling history can be easily maintained at vehicles, and communicated to the current AP at the beginning of the local production phase.

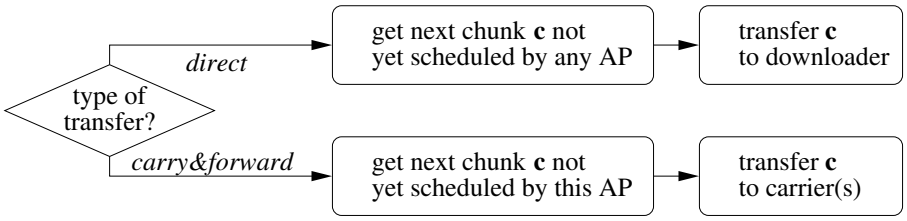


Figure 3.10: Flow diagram of the Hybrid chunk scheduling algorithms

The **Local** chunk scheduling is similar to the Hybrid scheme, since different APs can schedule the same chunks when delegating data to carriers. However, as shown in Fig. 3.11, it also allows overlapping between direct and carry&forward transfers. An AP can thus directly transfer to a downloader within range chunks that were already scheduled, but through a carry&forward delivery. Namely, an AP can employ a direct transfer to a downloader car to fill gaps in its chunk list. The Local scheduling is thus the most redundant among the schemes we propose, trading some cooperative download potential for increased reliability in data delivery.

3.4 Evaluation scenarios

In order to evaluate the cooperative download mechanisms outlined in the previous sections, we consider several large-scale vehicular traffic scenarios, that are representa-

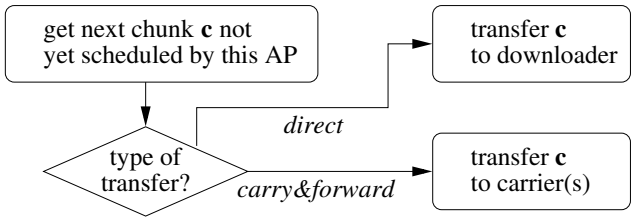


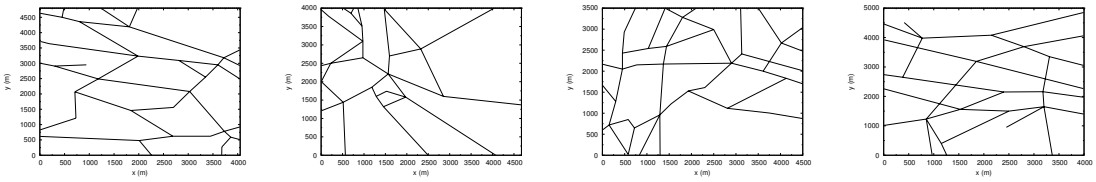
Figure 3.11: Flow diagram of the Local chunk scheduling algorithms

tive of real-world road topologies. We also take into account different deployments of APs, that, as we will show, have a major impact on the download performance. In the following, we first present the mobility scenarios and then discuss the APs deployment strategies employed in our performance analysis.

3.4.1 Vehicular mobility

We selected real-world road topologies from the area of Zurich, Switzerland, to assess the performance of the cooperative download solutions presented in the previous sections. This choice was mainly driven by the availability of large-scale microscopic-level traces of the vehicular mobility in the region, from the CS Department of ETH Zurich (82). The simulation techniques and mobility models employed to generate the traces allow to reproduce vehicular movements over very large road topologies, yet with a good degree of precision (83). More precisely, the traces replicate macroscopic patterns of real-world vehicular traffic flows, made up of thousand of cars, as well as microscopic behaviors of individual drivers in urban environments, such as pauses at intersections that depend on roads capacity and congestion. This macro- and micro-mobility realism is important in our study, since, on the one hand, we exploit large-scale properties of urban vehicular mobility in designing the cooperative download system, and, on the other, realistic small-scale mobility is required to reproduce vehicle-to-vehicle and vehicle-to-AP networking interactions. We stress that the mobility traces we employed only reproduce important traffic arteries in each scenario, while they do not consider movements over minor streets. This, however, does not impact our study, since the traffic over minor roads is too sporadic to deserve the deployment of dedicated APs, and does not provide significant opportunities for collaboration among vehicles.

We focused on four scenarios, representing urban, suburban, and rural areas within and nearby the city of Zurich. All the areas considered cover surfaces between 15 and



(a) West Zurich (b) Central Zurich (c) North Zurich (d) Schlieren

Figure 3.12: The road topologies considered in our study, representing urban (b), suburban (a,c), and rural (d) areas in the canton of Zurich, Switzerland

20 km^2, and frame several tens of kilometers of major roads, whose layouts are shown in Fig. 3.12. In particular, the Central Zurich scenario reproduces the downtown of Zurich, and it is thus characterized by dense traffic uniformly distributed over the road layout. The West and North Zurich scenarios are representative of suburban areas, where the traffic congestion is less evident than in the city center, but still present over a few major freeways that attract most of the vehicular mobility. Finally, the Schlieren scenario portrays the street topology around a town not far from Zurich, characterized by a sparse presence of vehicles over most roads in the area.

3.4.2 AP deployment

The placement of APs over the urban road topology has a major influence on the cooperative download architecture. In order to capture such an effect, we extend our analysis by considering diverse AP deployments over the different road topologies presented above. The goal of all the deployment strategies is to position, along a road topology, a given number N of APs; in our performance evaluation, we will discuss the impact of the value of N as well.

Under the **Random** AP positioning scheme, each point of the road topology has the same probability of being selected for the deployment of an AP. The resulting placement may be considered representative of a completely unplanned infrastructure (84, 85), and it is used in our performance evaluation as a baseline for the other deployment techniques. We emphasize that the results we will present for the Random positioning scheme were obtained by generating different deployments at each simulation run, so to avoid biases due to more or less favorable random AP distributions.

The **Density-based** AP deployment technique aims at maximizing the probability of direct data transfers from APs to downloader vehicles. To that end, this techniques places the APs at those crossroads where the traffic is denser. The rationale behind such choice is that the volume of direct downloads is proportional to the number of APs that a downloader vehicle encounters during its movement through the road topology. Since the identity of downloaders cannot be known in advance, the best option is to deploy APs at those locations that a generic vehicle will most probably visit along its route, i.e., the most congested intersections.

The **Cross volume-based** AP placement is designed to favor carry&forward transfers, by increasing the potential for collaboration among vehicles. This technique exploits the predictability of large-scale urban vehicular traffic flows, which are known to follow common mobility patterns over a road topology (79, 80, 81). By studying such traffic dynamics, it is possible to determine the way vehicular flows spread over the streets layout and employ this information to guide the AP placement. In the remainder of this section, we introduce the concept of *cross volume* and employ it to determine the relative AP deployment strategy.

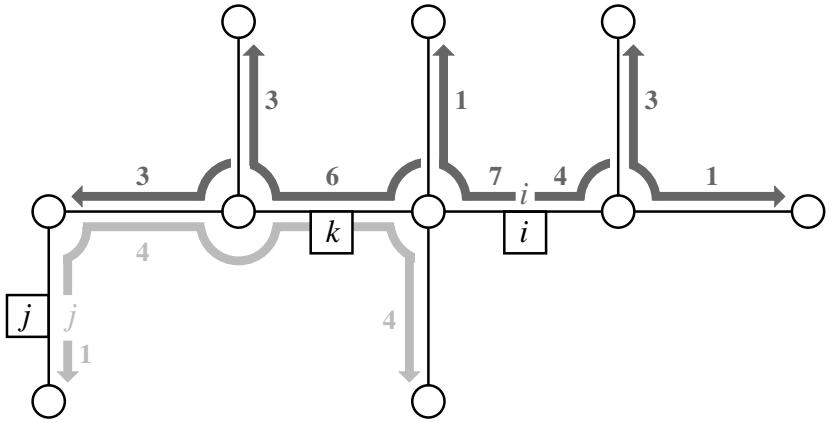


Figure 3.13: Sample vehicular flows over a road topology graph. Flows generated at edge i are dark grey, while those generated at j are light grey. Assuming a travel time = 1 at all edges, the partial cross volume h_{ij}^{k} is equal to $min\{6,4\} + min\{0,0\} = 4$, while the crossing volume h_{ij} is $4 + 3 = 7$

Let us imagine that the road topology is represented by a graph where vertices are mapped to intersections and edges to streets connecting them, as in Fig. 3.13. The graph is undirected, and an edge exists even if the corresponding road is one-way.

Focusing on a particular edge i of the graph, we can track all traffic leaving such edge[1], in both directions, and draw a map of how vehicular flows (measured in vehicles/s) from i unfold over the road topology. We refer to these flows as the vehicular flows *generated* at i. As an example, in Fig. 3.13, the dark grey arrows depict flows generated at edge i. Different flows have different size, in vehicles/s, represented by their associated number (values in the example are only illustrative).

Let us now consider a generic edge $k \neq i$, and isolate the flows passing in strict sequence through i and k. In this case, we distinguish the two directions of movement over k, and define two *traversing flows* from i to k:

- $\overrightarrow{f_{ik}}$, the vehicular flow generated at i and subsequently traversing k in the $\rightarrow$ direction;
- $\overleftarrow{f_{ik}}$, the vehicular flow generated at i and subsequently traversing k in the $\leftarrow$ direction.

We do not impose any rule in defining the two directions at k, which, e.g., could be based on vertices numbering or geographical coordinates. Our only concern is that, for each edge, the two directions are unambiguously identified. Also, the direction at i is not specified, and a traversing flow could have visited its generating edge in any direction (including both of them, if flows generated at i in opposite directions then merge at k in the same direction).

Traversing flows at an edge k can be translated to traversing volumes (measured in vehicles), by evaluating the average time vehicles spend to travel over the entire road segment corresponding to edge k. Also in this case, we can distinguish the two directions of movement, and define the two travel times $\overrightarrow{t_k}$ and $\overleftarrow{t_k}$, in the $\rightarrow$ and in the $\leftarrow$ directions, respectively. Considering again traversing flows from i, the two corresponding *traversing volumes* are

$$\overrightarrow{v_{ik}} = \overrightarrow{f_{ik}} \cdot \overrightarrow{t_k} \quad , \quad \overleftarrow{v_{ik}} = \overleftarrow{f_{ik}} \cdot \overleftarrow{t_k}$$

and represent the average number of vehicles that, having already visited i during their trip, travel over k in the $\rightarrow$ and $\leftarrow$ direction, respectively.

[1] Note that we do not make any assumption on the origin of the traffic, that could thus be constituted of vehicles that started their trip from an intermediate point of the road, or that had previously arrived from a different road.

Let us now introduce a second group of flows, generated at an edge $j \neq i$, depicted in light grey in Fig. 3.13. The same consideration we made for flows generated at i are valid, and, picked an edge $k \neq j$, we can compute the traversing volumes from j to k, $\overrightarrow{v_{jk}}$ and $\overleftarrow{v_{jk}}$. By considering both sets of flows at once, we can define the *partial cross volume* of i and j at k, as

$$h_{ij}^{k} = \begin{cases} min\left\{\overrightarrow{v_{ik}}, \overleftarrow{v_{jk}}\right\} + min\left\{\overleftarrow{v_{ik}}, \overrightarrow{v_{jk}}\right\}, & \text{if } k \neq i, k \neq j \\ 0, & \text{otherwise.} \end{cases}$$

The partial cross volume h_{ij}^{k} corresponds to the amount of traffic from i and j that merges at edge k. We notice that h_{ij}^{k} only couples flows that travel on opposite directions over k, hence the name of partial *cross* volume. The rationale is the following: imagine one car that has visited edge i in its trip, and now enters edge k in the $\rightarrow$ direction: such a car thus belongs to the $\overrightarrow{f_{ik}}$ flow. Considering vehicles that come from edge j and now travel over k, our car can generate two types of contacts:

- with the $\overleftarrow{v_{jk}}$ vehicles that travel in the opposite direction. These contacts are certain, since u-turn are not allowed on road segments connecting two adjacent intersections;
- with the $\overrightarrow{v_{jk}}$ vehicles that travel in the same direction. However, contacts are not certain in this case: the relative speed is close to zero[1] and contacts mostly depend on the position of the $\overrightarrow{v_{jk}}$ vehicles over k, when our car enters the road segment. Indeed, even if it enters edge k while some of the $\overrightarrow{v_{jk}}$ vehicles are nearby, and thus generates contacts with them, our car will only meet that very small fraction of the overall $\overrightarrow{v_{jk}}$ vehicles.

Since there are $\overrightarrow{v_{ik}}$ cars such as the one considered above, we couple $\overrightarrow{v_{ik}}$ with $\overleftarrow{v_{jk}}$, as these volumes correspond to certain contacts, while we do not couple $\overrightarrow{v_{ik}}$ with $\overrightarrow{v_{jk}}$, as these volumes have an unpredictable (and, in most cases, negligible) contribution in terms of contacts. Such a coupling is performed by taking the minimum between the two facing traffic volumes, which is that imposing a more strict constraint on the number of encounters.

[1] In the urban, suburban and rural scenarios we consider, the low speed limits and the reduced number of lanes hinder overtakings. This is proved by the fact that, in the scenarios in Sec. 3.4.1, we observed, on average, less than one overtaking per vehicle and per trip.

Finally, the concept of cross volume can be unbound from intermediate edges and related to couples of roads only. If I is the set of edges in the road topology graph, we define the *cross volume* of i and j as

$$h_{ij} = \begin{cases} \sum_{k \in I} h_{ij}^k, & \text{if } i \neq j \\ 0, & \text{otherwise,} \end{cases} \tag{3.1}$$

which implies that $h_{ij} = h_{ji} \geq 0$, $\forall i, j \in I$. The cross volume h_{ij} provides a measure of the potential for contact, and thus cooperation, over the entire road network, among vehicles leaving edges i and j. We can exploit such a measure to formalize the problem of the AP deployment. Let us assume that there are $|I| > N$ roads in the topology: the problem becomes that of picking N out of the $|I|$ edges of the graph for AP deployment. We associate to each edge i a binary decision variable x_i:

$$x_i = \begin{cases} 1, & \text{if an AP is deployed on the road mapped to } i \\ 0, & \text{otherwise,} \end{cases}$$

and refer to their vector as $x = \{x_1, \ldots, x_{|I|}\}$. Since opportunities for cooperation between vehicles are proportional to the crossing volume between each couple of edges, the APs should be positioned so to maximize the sum of crossing volumes between each pair of APs over the whole road topology. This leads to the formulation of the following mixed-integer quadratic programming (MIQP) problem:

$$\max_x \quad f(x) = \frac{1}{2} x' H x \tag{3.2}$$

$$s.t. \quad x_i \in \{0, 1\}, \quad \forall i \in I \tag{3.3}$$

$$\sum_{i \in I} x_i \leq N. \tag{3.4}$$

$$x_i + x_j \leq 1, \quad \forall j \in \Omega_i,\ j < i,\ \forall i \in I. \tag{3.5}$$

Here, that in Eq. 3.2 is the objective function to be maximized, with $H = \{h_{ij}\}$ being a $|I| \times |I|$ matrix in $\mathbb{R}$, filled with the crossing volumes computed for each couple of edges as in Eq. 3.1. From the definition of crossing volume, $h_{ij} = h_{ji} \geq 0$, $\forall i \in I$, and $h_{ii} = 0$, $\forall i \in I$, which guarantees that the matrix is symmetric with trace zero, thus it is semidefinite positive and the quadratic programming problem is convex. Also, Eq. 3.3 states that x_i is a binary variable, $\forall i \in I$, whereas Eq. 3.4 bounds the overall number of APs to be deployed to N.

The constraint in Eq. 3.5 avoids that two adjacent edges (i.e., edges with a vertex in common) both host APs. As a matter of fact, Ω_i is the set of indices j labeling edges that are adjacent to edge i: the inequality thus forces the sum of any two decision variables x_i and x_j referring to adjacent edges to be less than or equal to 1. This prevents that APs are placed too close to each other and form locally dense clusters, where only direct communication is used and insufficient space is left in between APs to perform vehicle-to-vehicle transmissions. For the same reason, we only deploy one AP per road.

By solving the optimization problem, we obtain an AP deployment that, as originally stated, augments the opportunities for carry&forward transfers in the download process. We note that this formulation solves the AP deployment problem from a large-scale viewpoint, i.e., it allows to determine the roads where APs should be positioned. However, it does not specifies the exact location of each AP over the selected roads. In Sec. 3.5.2, we will show that such small-scale deployment has a negligible impact on the performance of the system.

The problem can be easily extended to the case of incremental AP deployment, in which one or more APs have already been positioned along some roads, and others are to be added. It is sufficient to introduce constraints that force x_i to 1, for all edges i where an AP is already deployed.

3.5 Performance evaluation

We conducted an extensive simulation campaign aimed at evaluating multiple aspects of cooperative download in non-highway vehicular networks. The computational complexity of the simulations, that reproduce the movement and network traffic of several thousands of vehicles at a time, prevented the use of a traditional network simulator, such as *ns-2*. Instead, we developed a dedicated simulator (86), which employs the mobility extracted from the Zurich traces, models a random access channel contention, and implements all the cooperative download techniques previously presented, but replaces the traditional packet-level simulation with a more scalable chunk-level one, avoiding the detailed reproduction of the entire network stack at each node.

In all our tests, a "best case" performance reference is provided by an **Oracle** carriers selection algorithm. The Oracle scheme assumes that APs have a perfect

knowledge of the future trajectories of all vehicles, in terms of both routes and timings. This information is exploited during production phases at APs to foresee contacts between local and downloader vehicles, and thus to pick carriers that are certain to later meet their target downloader vehicle. The Oracle scheme exploits a per-downloader database, identical to that employed in the (p,t)-Constrained algorithm, to avoid that multiple carry&forward transfers are scheduled at a same time for the same downloader vehicle. Also, since vehicular contacts are known in advance, any redundancy in the scheduling of chunks is pointless: thus, only one carrier can be selected for the transfer of each carry&forward chunk, and we always couple the Oracle algorithm with the non-redundant Global chunk scheduling. Note that the Oracle carriers selection algorithm is not optimal in that it does not take into account that multiple carry&forward transfers can be scheduled at the same time for different downloaders that are within transmission range of each other. In such situation, since only one of the interfering downloaders can receive its chunks, part of the data cannot be delivered to the second downloader.

The main metrics we are interested in evaluating are:

- the *download rate*, i.e., the average file transfer speed experienced by downloader vehicles traveling through the scenario. Such rate is the aggregate of a *direct* rate, due to direct data downloads from APs, and a *cooperative* rate, due to carry&forward transfers. According to our simulation settings, listed next, the maximum download rate achievable by a vehicular user is 5 Mbps, which corresponds to the case of a car continuously receiving data during its whole trip through the simulation scenario;

- the *undelivered chunk ratio*, i.e., the average ratio of chunks that are not delivered to a downloader vehicle, computed over all those scheduled for that vehicle.

The system parameters were set for all simulations to $\delta t = 5s$, $\delta\alpha = 45°$, $\delta v = 5\frac{m}{s}$, $T = 500s$, $\mathbb{T} = 1s$, $\mathbb{P}_{min} = 2.5$, $\mathbb{P}_{ind} = 0.5$. If not stated otherwise, an average of ten downloader cars is present at the same time over the road topology, and a simple disc model is considered for signal propagation, so to fulfill the low complexity constraints imposed by the size of the simulations. The net application-level data transmission rate over the wireless channel, during both production and forward phases, is assumed equal to 5 Mbps for a transmission range of 100 m. The latter values are consistent with the outcome of real-world experiments on car-to-infrastructure (87) and car-to-car

data transfers (88). For each simulation we performed one run to train the framework (i.e., gather the data necessary to deploy the APs and to build the contacts map at each APs), and ten runs to collect statistics. Each run simulated around three hours of urban traffic, encompassing various vehicular density conditions. For all results we measured 99% confidence intervals, reported as error bars in the plots.

3.5.1 Carriers selection and chunk scheduling

We first analyze the different carriers selection algorithms and chunk scheduling techniques, detailed in Sec. 3.3.1 and Sec. 3.3.2. Since we are interested in a comparative evaluation of the all these schemes, we select a particular AP deployment scenario (6 APs positioned according to the Cross volume-based strategy), and focus on the carry&forward download performance (as direct downloads are not influenced by they way we select carriers or chunks). We will consider different AP deployments, and study their impact on direct download rates in Sec. 3.5.2.

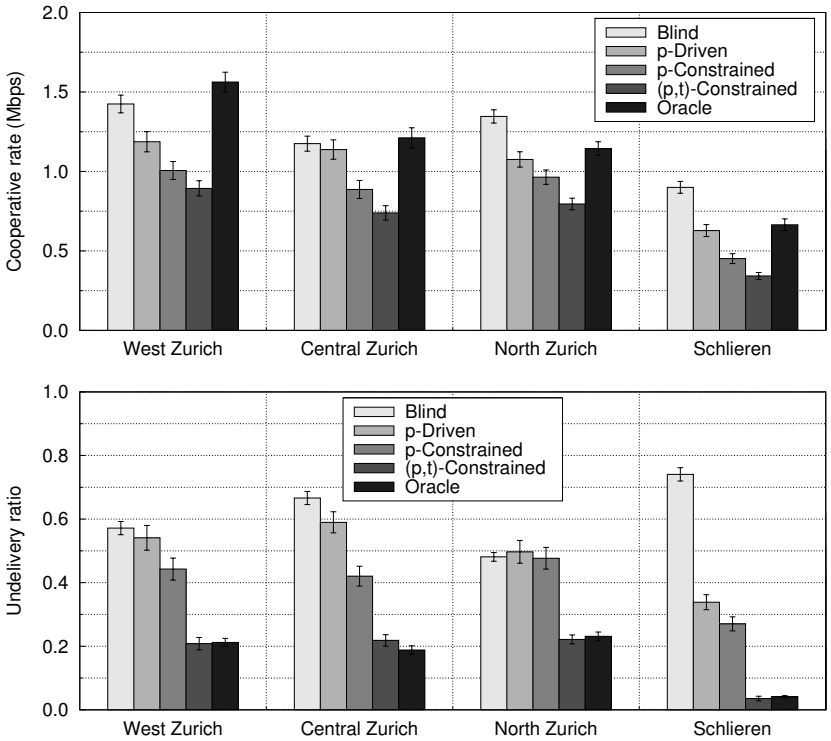


Figure 3.14: Average cooperative download rate (top) and undelivery ratio (bottom) for different carriers selection schemes, in the four road topologies

The average cooperative download rate, obtained from carry&forward transfers, and the mean undelivery ratio are depicted in Fig. 3.14. There, we report the results obtained under each road scenario by the different carrier selection scheme, when coupled with a Global chunk scheduling. We can notice how the Blind, p-Driven, p-Constrained, and (p,t)-Constrained algorithms yield, in this order and throughout all road scenarios, decreasing cooperative rates, as well as reduced undelivery ratios. Indeed, increasing the precision of carry&forward transfers also implies missing opportunities for vehicle-to-vehicle data exchanges.

The exact balance in the tradeoff between the cooperative download rate and the delivery precision varies with the road scenario. In suburban areas (West and North Zurich), a few major freeways attract most of the traffic in the region. On the one hand, such concentration results in a large number of vehicle-to-vehicle contacts, and thus in higher cooperative rates with respect to other scenarios. On the other hand, it favors inaccurate carriers selection algorithms, such as the Blind one, over more precise schemes, such as the p-Driven and p-Constrained ones: as a matter of fact, even randomly selected cars have a high probability of traveling on the same road, and thus to meet each other. In the rural Schlieren scenario, the sparsity of traffic leads to reduced car-to-car contacts and thus lower cooperative rates. Also, the higher heterogeneity in the movement of vehicles, due to the absence of traffic-gathering roadways, makes the Blind scheme extremely imprecise in delivering chunks, with respect to contact map-based ones. Finally, the urban Central Zurich scenario presents traffic densities that are similar to those observable in the suburban areas, but dynamics that are closer to those of the rural case: such scenario thus yield high cooperative rates but undelivery ratios that significantly vary under the diverse algorithms.

However, we can notice that, no matter the road scenario considered, the (p,t)-Constrained carriers selection algorithm significantly outperforms all the other solutions in terms of undelivery ratio. Indeed, the precision achieved by the (p,t)-Constrained algorithm in the carry&forward chunk delivery is comparable to that of the Oracle algorithm. Although this latter scheme attains higher cooperative rates, we can state that the overall performance of the (p,t)-Constrained algorithm is not too far from that obtained through a perfect knowledge of future contacts among vehicles.

In Fig. 3.15, we also report, for the same combinations of carriers selection algorithms and road scenarios, the average load measured at the APs, i.e., the percentage

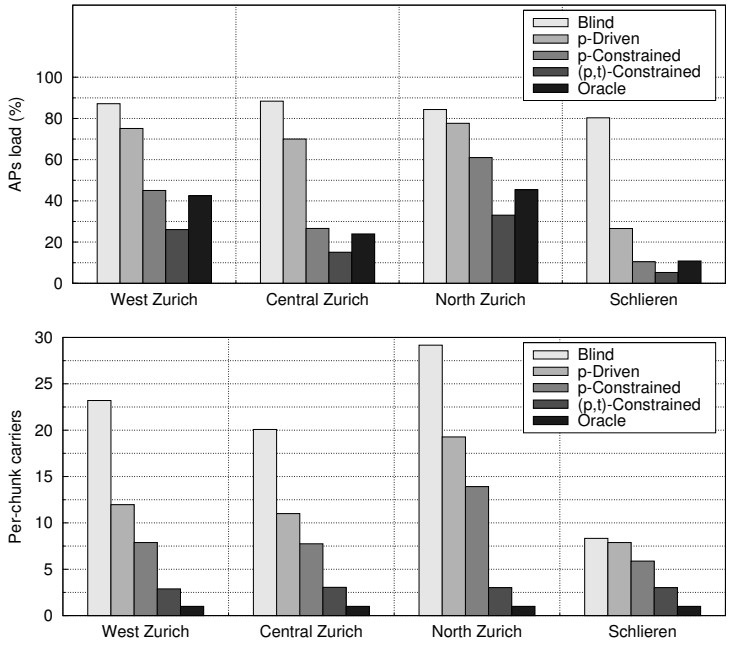


Figure 3.15: Average AP load (top) and number of carriers ferrying a same chunk (bottom) for different carriers selection schemes, in the four road topologies

of airtime used by an AP to transfer data to carriers, and the mean number of carriers ferrying a same chunk to a target downloader. From the plots it is clear how more precise algorithms result in a lower AP load and a smaller number of carriers per chunk. In other words, a higher precision in the carry&forward delivery of chunks yields a lower charge on the infrastructure and a reduced demand of resources by cooperating vehicles.

The different chunk scheduling schemes are then compared, in combination with every carriers selection algorithm, in Fig. 3.16. For the sake of brevity, the results are aggregated over all four road topologies: indeed, the same behaviors we previously discussed for the diverse mobility scenarios were observed also in this case. As a general comment, the increased redundancy introduced by the Hybrid and Local chunk scheduling leads, as one could expect, to lower cooperative rates but increased delivery precision. On a per-carriers selection algorithm basis, however, differences can be spotted. In particular, the Blind scheme suffers a dramatic 50% reduction in the cooperative rate when redundancy is increased in the chunk scheduling: indeed, scheduling multiple times the same chunks impairs the major strength of the Blind algorithm, i.e.,

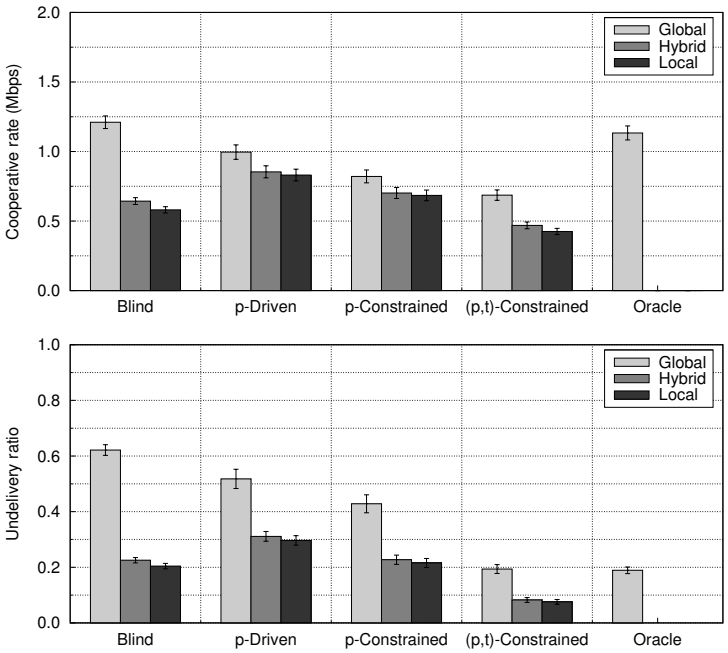
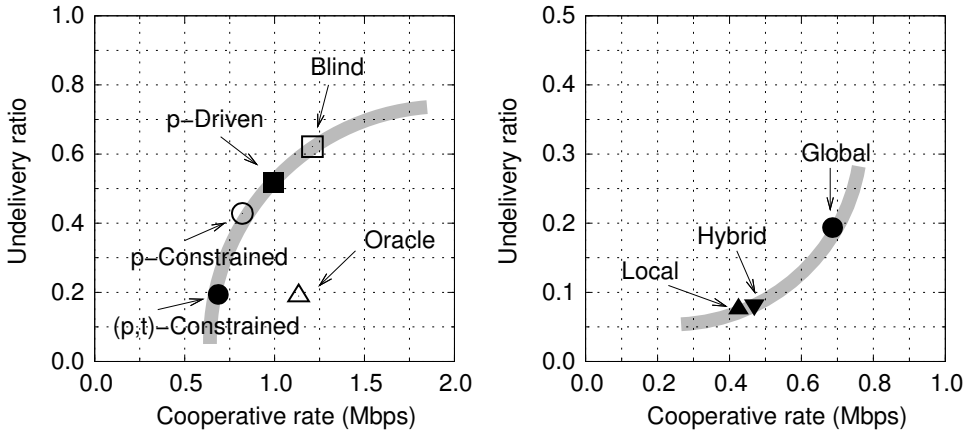


Figure 3.16: Average cooperative download rate (top) and undelivery ratio (bottom) for different chunk scheduling algorithms, under the diverse carriers selection schemes. Results are averaged over all road topologies



(a) Carriers selection

(b) Chunk scheduling

Figure 3.17: Cooperative download rate / undelivery ratio working points for different carriers selection schemes (left, with Global chunk scheduling) and chunk scheduling algorithms (right, with (p,t)-Constrained carriers selection). Results are averaged over all road topologies

the sheer number of unique chunks sent out for randomly selected downloaders. Conversely, when coupled with more redundant chunk schedulings, the algorithms based on contact maps enjoy a significant reduction in the undelivery ratio (from 35% to 65%, depending on the algorithm) at some smaller cost (15% to 30%) in terms of cooperative rate.

To conclude our analysis on carriers selection and chunk scheduling, we summarize the results in Fig. 3.17, showing the working points of each technique in the download rate/undelivery ratio space. The results, aggregated over all road topologies, evidence the tradeoff between the download volume and the reliability of the cooperative process. In particular, the non-linear distribution of the working points, evidenced by the grey curves in the plot, seem to indicate the (p,t)-Constrained carriers selection with Global chunk scheduling as the combination that better adapts to the different scenarios.

3.5.2 Impact of the AP deployment

The way the infrastructure is deployed can have a significant impact on the performance of the cooperative download framework. Thus, in this section, we evaluate how the strategy employed for the AP placement and the number of fixed stations influence the

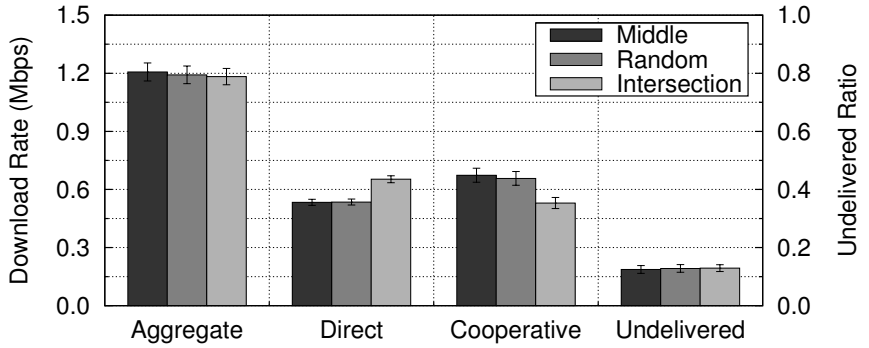


Figure 3.18: Download rates and undelivered chunks ratios for different road-level placement policies, averaged over all road topologies

rates and undelivery ratios experienced by the downloader vehicles. In the light of the results in the previous section, we consider in the following a (p,t)-Constrained carriers selection with Global scheduling as our default configuration.

We initially demonstrate the negligible impact of the small-scale deployment of APs under the Cross volume-based strategy outlined in Sec. 3.4.2. To that end, we compare three policies for the placement of APs over the roads resulting from the optimization problem. The *middle* policy places an AP equidistant from the intersections that end the selected road segment, the *intersection* policy deploys an AP at the most crowded of such intersections, while the *random* policy picks a random location over the selected road segment. Fig. 3.18 shows that the relevance of road-level deployment is minimal, as the three schemes achieve almost identical download rate and undelivered chunk ratio. The only notable difference is in that the intersection strategy favors direct downloads and penalizes cooperative ones: indeed, crossroads are characterized by high densities of slow vehicles, and placing APs there favors AP-to-vehicle transfers. At the same time, however, it deprives vehicles of transfer opportunities, since intersections also represent network clustering points where car-to-car contacts occur frequently (45), thus reducing the cooperative download rate. We thus consider APs to be deployed at the intermediate point of road segments, as this appears to bring a slight advantage over the other policies in terms of aggregate rate.

The different AP deployment strategies discussed in Sec. 3.4.2 are compared in Fig. 3.19, in presence of 6 APs deployed in each road scenario. It is clear that a Random AP deployment results in the worst performance, as both the direct rate, i.e.,

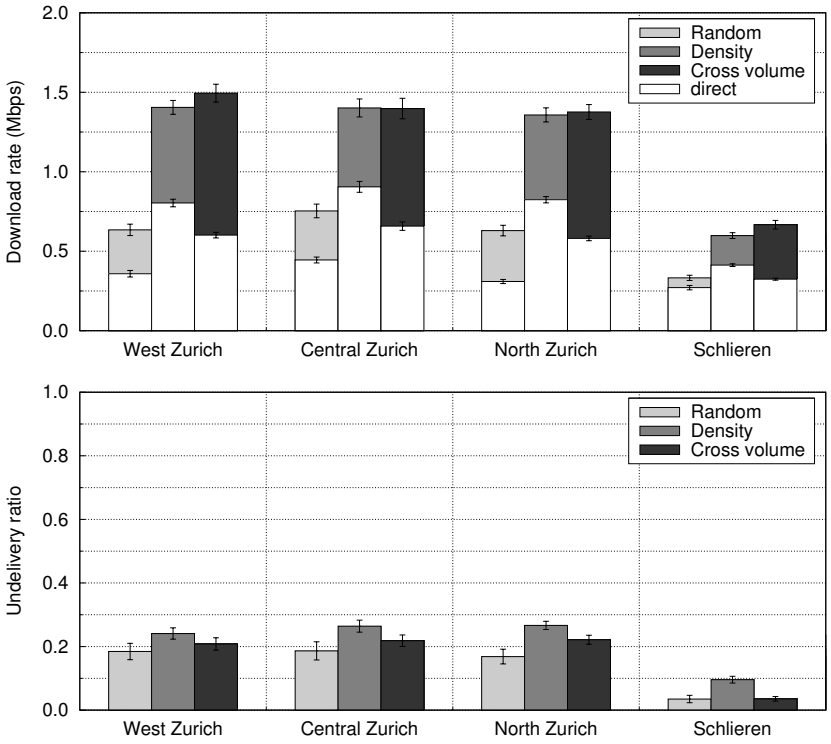


Figure 3.19: Average download rates (top) and undelivery ratio (bottom) for different AP deployment strategies, in the four road topologies. Results refer to the (p,t)-Constrained carriers selection scheme

the portion of the total download rate due to chunks directly retrieved from APs, and the cooperative rate, resulting instead from carry&forward transfers, are lower than in the other strategies. The reduced aggregate rate does not even bring an advantage in terms of undelivery ratio, which is comparable to that obtained under other deployments. Conversely, the Density-based and the Cross volume-based AP placement lead to similar aggregate rates: however, in the former the contribution of direct transfers is significantly larger than in the latter, that instead mainly leverages the cooperation among vehicles. Such result is consistent with the objectives of the two strategies, that try to maximize, respectively, vehicle-to-infrastructure and vehicle-to-vehicle contacts. Interestingly, the Density-based strategy yields slightly higher undelivery ratios with respect to the other deployments: as already discussed, deploying APs at intersections renders many opportunistic contacts among vehicles unusable for planned data exchanges, an effect here exacerbated by the high densities of the junctions selected by

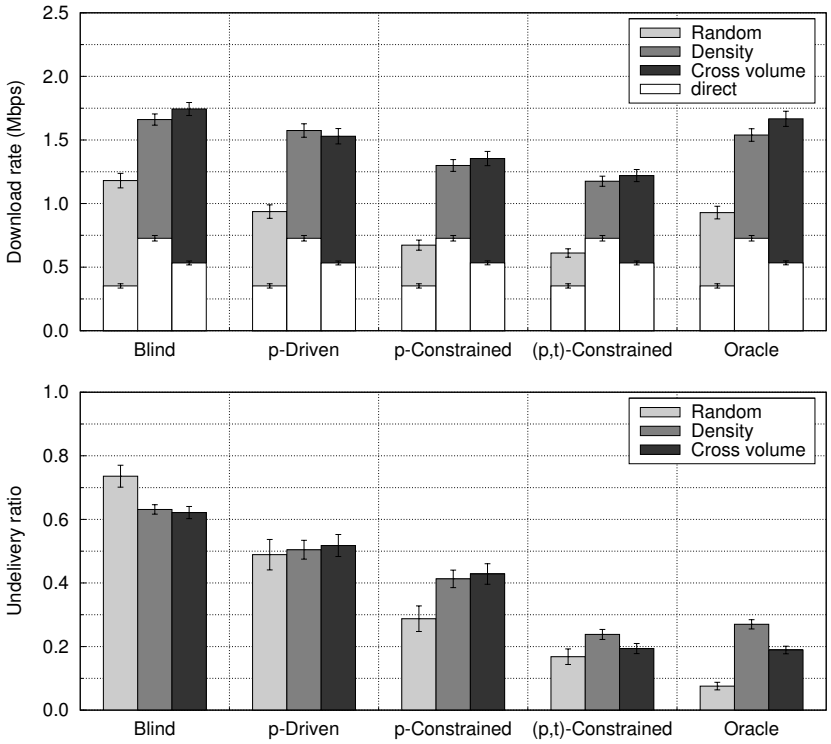
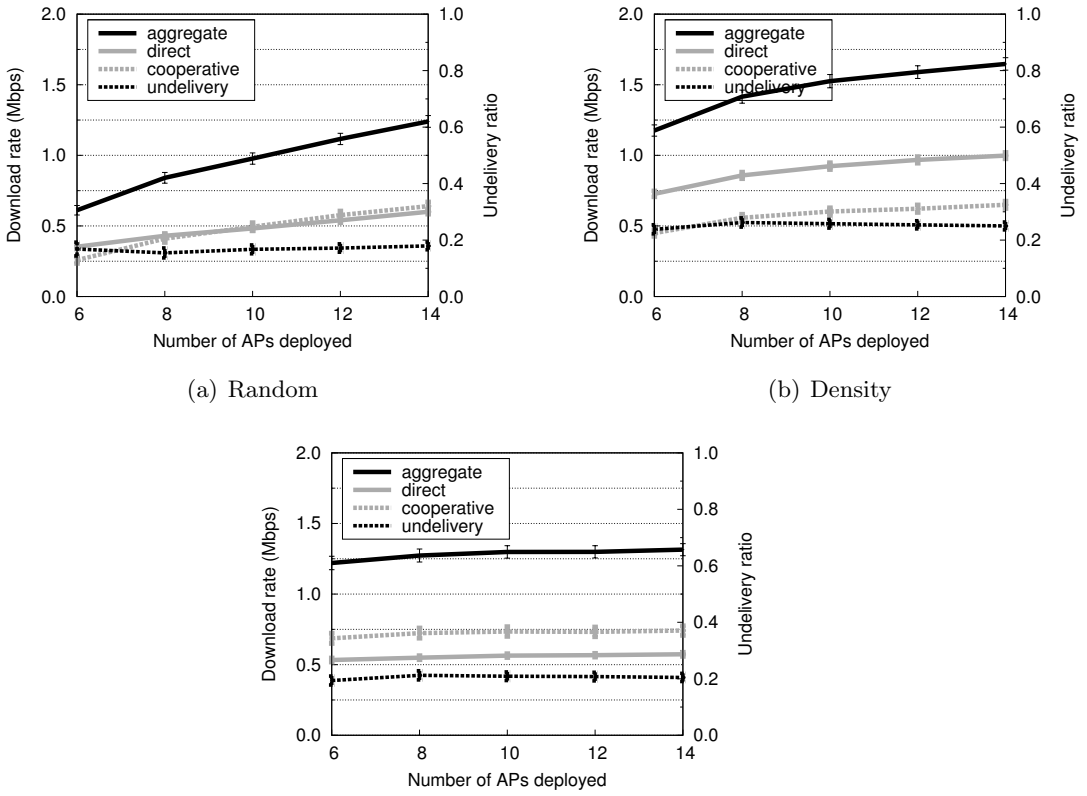


Figure 3.20: Average download rates (top) and undelivery ratio (bottom) for different AP deployment strategies, under the diverse carriers selection algorithms. Results are averaged over all road topologies

the Density-based deployment.

Focusing on the proportion between direct and cooperative rates, we can observe that, depending on the road topology and deployment scenario, the carry&forward contribution typically varies between 35% and 60% of the total download rate, implying a remarkable 50% to 120% speedup in the download. As far as the road topologies are concerned, the same considerations made in the previous sections hold for the overall rates and undelivery ratios. Moreover, different scenarios do not appear to induce significant differences in the relative performance of each deployment, nor in terms of the proportion between direct and cooperative rates.

One may wonder how different AP placements affect carriers selection algorithms other than the (p,t)-Constrained one. In Fig. 3.20 we can observe that the strategy adopted in the deployment of the infrastructure has a very similar influence on all the algorithms. The cooperative rates achieved by the diverse schemes are consistent with



(a) Random

(b) Density

(c) Cross volume

Figure 3.21: Average download rates and undelivery ratio for a varying number of APs, under the diverse deployment strategies. Results refer to the (p,t)-Constrained carriers selection scheme and are averaged over all road topologies

those presented in the previous section, and thus have an even higher impact on the aggregate rate, with respect to the case of the (p,t)-Constrained algorithm studied above. Such an improvement comes, however, at a high cost in terms of undelivered chunks, with the exception of the Oracle scheme, which is clearly favored by its preemptive knowledge of future contacts.

Not only the position, but also the number of the APs can impact the performance of the vehicular cooperative download. In Fig. 3.21, we vary the number of APs deployed in each scenario, and show the average rates and undelivery ratio attained under the three AP placement strategies. Under a Random deployment, both direct and cooperative rates significantly increase as the infrastructure becomes more

pervasive. Additional APs imply a higher number of direct transfer opportunities for the downloaders, which explains the improved direct rate. Similarly, denser APs are deployed closer to each other, making forward phases (i.e., contacts among vehicles) closer in time to production phases (i.e., contacts between vehicles and APs) and thus easier to predict.

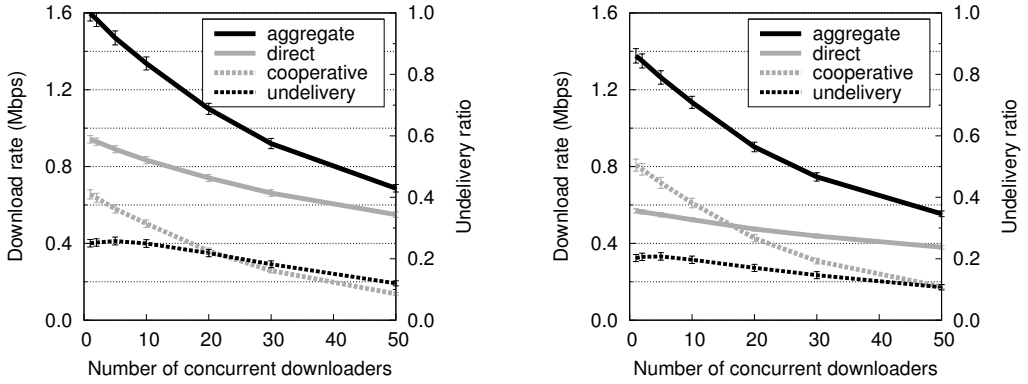
The same behavior can be observed for the Density-based strategy, which, however, shows a slower rates growth for high numbers of APs. This is explained by the fact that, while in the Random deployment each new AP has a similar impact on the download process, in the Density-based case additional APs are located at intersections characterized by decreasing vehicular densities. Therefore, the rate gain brought by the presence of extra APs tends to be lower and lower. In the case of APs positioned according the Cross volume-based policy, the rates increase is significantly lower than in the other cases. As a matter of fact, the optimization problem behind this placement strategy struggles to find new locations that guarantee an increase in the crossing volumes, and thus picks positions that introduce very small gain in the download process. That is, the same reasoning made for the Density-based deployment holds, exacerbated, in the Cross volume-based case.

Finally, it is interesting to note that, under all deployments, as the number of APs grows the ratio between direct and cooperative rates is either unchanged or slightly shifted in favor of the latter. Moreover, the undelivery ratio remains constant. Thus, we can conclude that the positive impact of carry&forward transfers on the download process persists as the number of APs deployed on the road topology varies.

3.5.3 Scalability in the number of downloaders

We evaluate the scalability of the cooperative download framework by increasing the number of downloaders concurrently traveling over each scenario, up to 50. This last value corresponds to a stressed network, where the vehicles actively downloading large-sized contents from the Internet jointly cover around one third of the whole road surface with their transmission ranges. Such a condition creates a significative contention for the channel, since it is very probable for two downloaders to interfere with each other, and thus impairs both direct and cooperative download.

This notwithstanding, in Fig. 3.22 we can observe that the system scales quite well, as each of 50 simultaneous users experiences an average aggregate rate that is only



(a) Density

(b) Cross volume

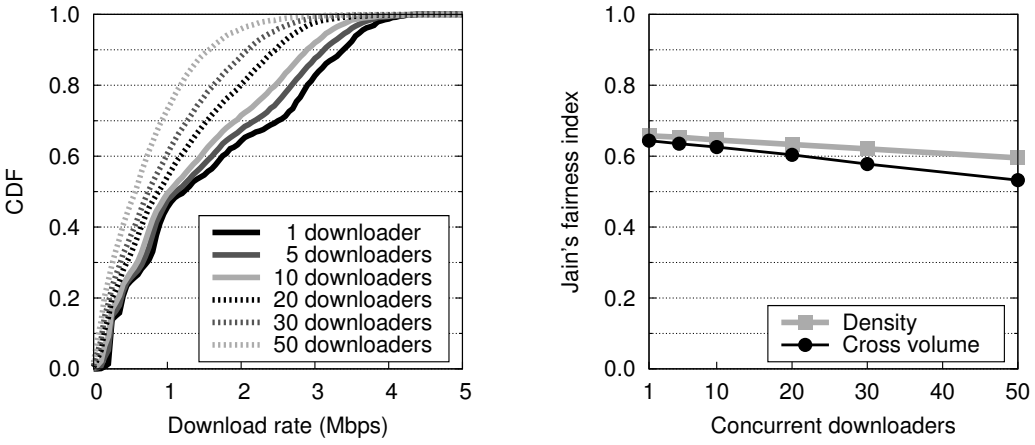
Figure 3.22: Average download rates and undelivery ratio for a varying number of concurrent downloaders and 10 APs. Results refer to the (p,t)-Constrained carriers selection scheme and are averaged over all road topologies

one half of the rate enjoyed by a lone downloader. We can also notice that cooperative transfers are affected more significantly than direct ones: since direct transfers always have a higher priority over carry&forward ones, the increasing presence of direct downloaders reduces the airtime that APs can dedicate to carriers, thus limiting the exploitation of the latter paradigm. The undelivery chunk ratio is instead positively affected by the downloaders' density: by increasing the spectrum of targets, it is easier for the APs to find one downloader that will encounter the local carriers with high probability.

3.5.4 Per-downloader performance analysis

One legitimate question at this point would be if these average figures are representative of the experience of every downloader. By looking at Fig. 3.23(a), the answer seems to be no. Indeed, the cumulative distribution of the aggregate download rate shows a significative unfairness among downloaders: as an example, in presence of a Density-based AP deployment[1] and 10 concurrent downloaders, the least fortunate 30% of the downloaders gets a goodput of at most 700 Kbps, while the top 30% can retrieve the desired content at a rate that is at least four times higher. To better capture such unfairness, in Fig. 3.23(b) we portray the Jain's fairness index associated to the

[1]Similar results were obtained under the Cross volume-based deployment.



(a) Download rate PDF

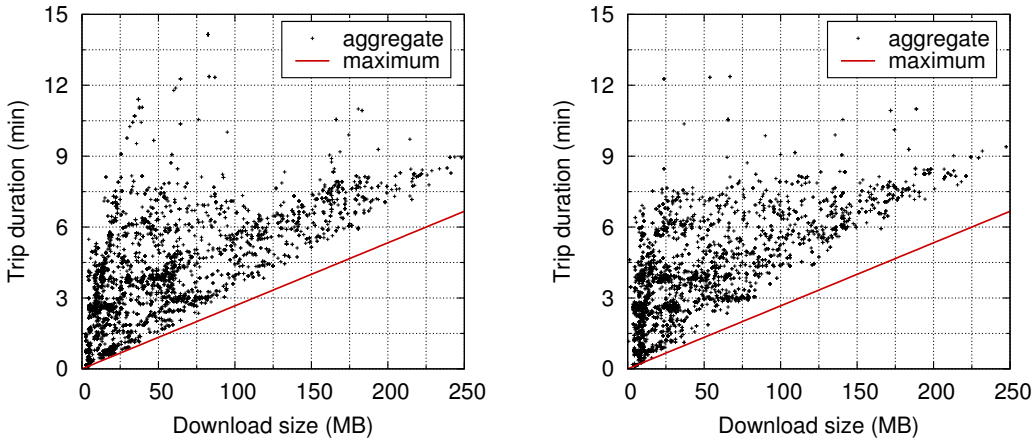
(b) Jain's fairness index

Figure 3.23: PDF of the aggregate download rate for a varying number of concurrent downloaders (left, for the Density-based deployment) and associated fairness (right). Results refer to the (p,t)-Constrained carriers selection scheme in presence of 10 APs, and are averaged over all road topologies

distributions of Fig. 3.23(a). We can notice that the index ranges between 0.5 and 0.7, i.e., low values that confirm the scarce equity with which the overall download rate is distributed among downloaders. The number of concurrent downloaders appears instead to have a minor impact on the fairness, as it only induces a slight reduction of the index.

In order to understand the reason of the diverse download experience of the different users, we first observe if there is a correlation between the amount of downloaded data and the duration of the trip of a vehicle, intended as the interval between the instants at which the car enters and exits the road scenario. The relative scatterplots are depicted in Fig. 3.24, where we also report the line representing the maximum downloadable file size per trip length, computed as the 5 Mbps data rate times the trip duration. Interestingly, we can observe that download sizes close to the maximum are attained by vehicles with short as well as long trips. Moreover, downloaders engaged in medium-to-long trips can have very different download experiences, resulting in both very high and very low download sizes. Our conclusion is that the duration of the trip is not the cause behind the unfairness observed in the download performance of different users.

We then increase the level of detail of our analysis, and consider not just the duration



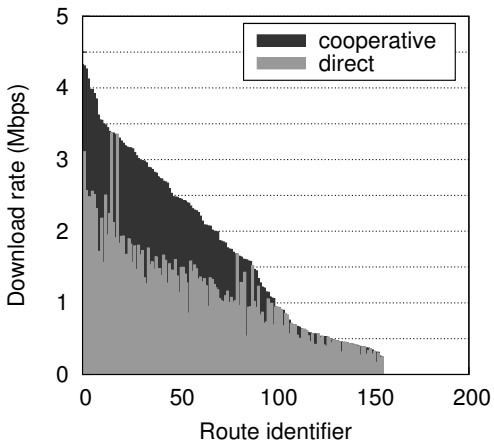
(a) Density

(b) Cross volume

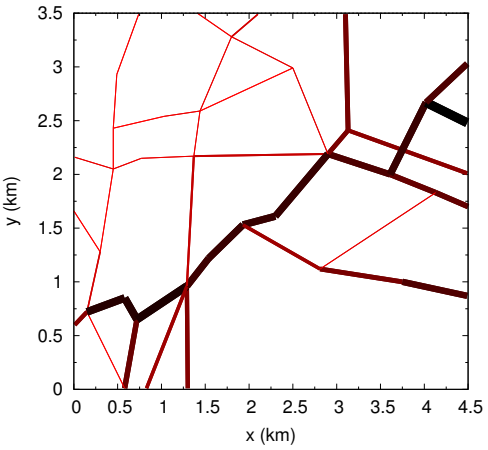
Figure 3.24: Scatterplot of the downloaded size versus the trip duration. Results refer to the (p,t)-Constrained carriers selection scheme in presence of 10 APs, and are averaged over all road topologies

of a trip, but its exact trajectory. More precisely, we record all the possible routes (i.e., ordered sequences of roads) traveled by downloaders, and measure the average download rates experienced by users on each route. Fig. 3.25(a) and 3.25(c) show, for two sample scenarios[1], the direct and cooperative rate attained by vehicles traveling along the different routes, that are ordered over the x axis, by decreasing aggregate download rate. It appears now clear that the unfairness among downloaders is the result of an unfairness among trajectories, as some routes allow average rates of 3 Mbps or more, whereas others yield much poorer (e.g., 500 Kbps or less) performance. It is interesting to note that, no matter which AP deployment strategy and road scenario are considered, the routes that guarantee the highest rates are often those where carry&forward transfers are exploited the most. On the other hand, cooperation is completely absent on those trajectories that allow very low throughput. Fig. 3.25(b) and 3.25(d) complement the previous results, showing which routes provide higher download rates: the difference between lighter, thinner low-rate trajectories and darker, thicker high-rate ones is even more evident. We can therefore state that there are routes that are more keen to take advantage from a cooperative download framework, and others that are less so.

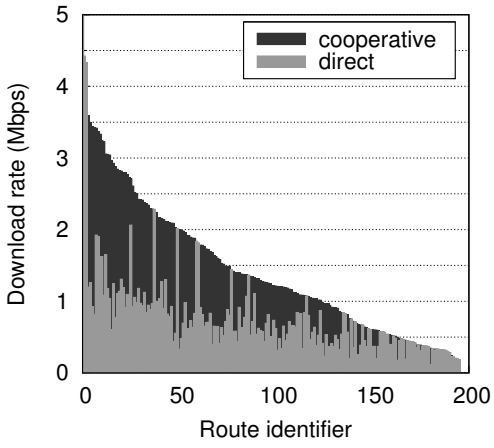
[1]Similar results, omitted for the sake of brevity, were obtained for all others combinations of AP deployment strategies and road topologies.



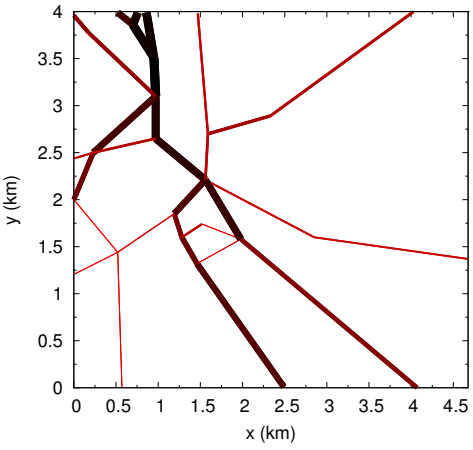
(a) Density, North Zurich, rates



(b) Density, North Zurich, map



(c) Cross, Central Zurich, rates



(d) Cross, Central Zurich, map

Figure 3.25: Extract of per-route analysis results, referring to the Density-based AP deployment in North Zurich (top) and to the Cross volume-based AP deployment in Central Zurich (bottom): in both cases, we report the average direct and cooperative rates over the possible vehicular trajectories, that are ordered by decreasing aggregate rate (left), and the download rates over the road topology, where darker and thicker lines represent road segments with higher aggregate rates. All results refer to the (p,t)-Constrained carriers selection scheme in presence of 10 APs

Scenario	Density	Cross volume
West Zurich	0.74	0.75
Central Zurich	0.57	0.64
North Zurich	0.59	0.63
Schlieren	0.42	0.60

Table 3.1: Correlation between the cooperative rate and vehicular density characterizing a same road, in different scenarios

Moreover, we also found a moderate positive correlation between the carry&forward download rate and the vehicular density that characterize a same road segment, as shown in Tab. 3.1: this suggests that it is over routes where the traffic is denser that cooperation among vehicles brings the highest advantage, thanks to a wider range of opportunistic contacts that can be leveraged for data transfers.

Our conclusion is that, if use of the carry&forward paradigm is judiciously limited to selected trafficked routes, cooperation among vehicles can provide significative and consistent increments in the download rates experienced by users traveling along such trajectories. The results in Fig. 3.25 indicate that, in such situations, one could expect typical gains in the order of 50% to 120% with respect to the case where only direct transfers from APs are considered.

3.6 Conclusions

We presented a complete study of cooperative download in urban vehicular environments. We identified and proposed solutions to the problems of carriers selection and chunk scheduling, and extensively evaluated them. The main contribution of this work lies in the demonstration that vehicular cooperative download in urban environments can bring significant download rate improvements to users traveling on trafficked roads in particular.

4

Exact Decoding Probability under Random Linear Network Coding

4.1 Introduction

Network coding allows efficient transmission from a set of sources to a set of destinations, allowing nodes to manipulate the information before forwarding it (11). It has been originally employed to improve the throughput of multicast transmissions over reliable links (11). Random linear network coding is a class of network coding, that operates on data through linear combinations of random codes (12). Although initially employed for multicast, random linear network coding has found wide application in networks with packet erasures, where it is used to improve the communication performance in absence of (or with limited resort to) feedback channels. Random linear network coding has been shown to improve the latency, capacity and energy efficiency of the communication in loss-prone and intermittently-connected wireless networks, either ad-hoc (13), delay-tolerant (14), or satellite and underwater (15).

Within these contexts, when one or more sources want to transmit N packets to one or more mobile nodes, the channel unreliability and the fluctuation of connectivity force the adoption of reliable communication techniques. However, traditional retransmission-based mechanisms easily lead to excessive overhead, even in presence of coordination between the sources. If random linear network coding is employed, the

reception of a (limited) amount S of excess packets can eliminate the need for a feedback channel. Indeed, to be able to decode the original data, a destination node simply has to acquire N linearly independent packets over the K=N+S it received from the source(s) and intermediate relay(s).

The probability of having vectors linearly independent does not depend on the topology of the network, but on the fact of how many vectors the receiver has in order to obtain a matrix of rank N. Note that the K packets can be encoded by any node as far as the N original packets are the same.

The effectiveness of random linear network coding thus depends on the the probability that at least N packets reach their destination, and that they represent linearly independent encodings of the original data. While the former condition relates to the error probability on the channel or to the network topology, on which one does not typically have control, the latter concerns the coding design, that is instead configurable. A common approach is to assume the size q of the Galois Field over which the coding is performed to be very large, so that any received packet is independent from those previously obtained, with high probability. As an example, using large encoding coefficients of 20 bits, that correspond to values of q in the order of 2^{20}, allows to exclusively dimension S on the packet loss probability. However, using large field sizes has a drawback in terms of computational complexity: it would thus be desirable to determine the exact impact of the field size on the decoding probability, so to properly dimension q. The only such result to date is the upper bound in (89), stating that the average number of coded packets K to be received before the original data can be decoded is, for a field size q, equal to

$$K = min\{N\frac{q}{q-1}, N+1+\frac{1-q^{-N+1}}{q-1}\}. \tag{4.1}$$

According to (4.1), when q=2, S=2 excess packets are sufficient, on average, for N+S received packets to be linearly independent, no matter the value of N. There, the authors derived an upper bound to the average number E of extra packets required to decode the original information, and showed how small values of q yield minimal increment in the value of E. More precisely, they show that, when q=2, E=2 excess packets are needed on average for an overall N+E received packets to be linearly independent.

In this thesis, we improve the upper bound in (4.1), by deriving the exact formulation of the probability that N out of $N+S$ received packets are linearly independent, under random linear network coding, as a function of the field size q. Our formulation evidences that a value of q equal to four allows a correct decoding with just one excess packet on average.

4.2 Exact decoding probability

Let $GF(q)$ be a Galois Field of size q and let us assume that a set of uncoordinated sources transmit K=N+S packets. Each k-th packet is constructed using random linear network coding; i.e., each new packet is associated with a random encoding vector g_k over $GF(q)$ of dimension N, and it is the result of the linear combination of the N original packets, (12). Let us call G_q the matrix containing the encoding vectors g_k. The N original packets can be decoded if G_q has rank N, i.e., the receiver node has obtained N linear independent packets over the K sent packets. We denote as P_{ns} the probability that matrix G_q has rank N.

In order to derive the exact expression of P_{ns} we employ an urn model. Consider an urn with all the vectors that can be generated by codes in a Galois Field of size q, $GF(q)$, over N packets. There are q^N possible vectors. For the sake of clarity, we will first analyze the simple case in which K=N, and then derive the case in which $K > N$.

Case $K = N$: consider the extraction of linearly independent vectors from the urn. In the first extraction any vector that is not the zero vector[1] will be a suitable vector. The probability of extracting a vector which is not the zero vector is equal to $(1 - \frac{q^0}{q^N}) = (1 - \frac{1}{q^N})$. After each extraction, we reinsert the extracted vector in the urn since in the model we are randomly extracting vectors over a Galois Field $GF(q)$. In the second extraction, there are q vectors that are dependent among them, so the probability of having two linearly independent vectors is $(1 - \frac{1}{q^N})(1 - \frac{q}{q^N})$. In the third extraction there are q^2 vectors that are linearly dependent with the previously extracted ones, and so on. As we perform exactly N extractions, and we need N independent vectors, we must not fail in any of the extractions. Thus, the probability of having N

[1]Discarding the zero vector is later treated.

linear independent vectors over K=N extractions is given by:

$$P_{ns}(K,N) = \prod_{j=0}^{N-1}(1-\frac{q^j}{q^N}) = \prod_{j=1}^{N}(1-\frac{1}{q^j}). \tag{4.2}$$

Case $K > N$: Let us first assume $K = N+1$ extractions from the urn. Again, during the first extraction, any vector that is not the zero vector is acceptable. However, in the second extraction, there is room for exactly one failure. If such failure occurs, with probability $\frac{q^0}{q^N}$, we will be left with exactly N-1 extractions to obtain N-1 independent vectors. If the newly extracted vector is instead independent with respect to the first one, with probability $(1-\frac{q}{q^N})$, we will still have N-1 extractions to get N-2 independent vectors. In the third extraction, we can fail with probability $\frac{q}{q^N}$ and must not fail with probability $(1-\frac{q^2}{q^N})$. Note that if a failure occurs in the *k-th* extraction, there must not be any failure in the future extractions, and thus these events are exclusive. Iterating, we obtain the probability that N linear independent vectors are extracted, given K=N+1 extractions, as:

$$P_{ns}(K,N) = \prod_{j=1}^{N}(1-\frac{1}{q^j})\sum_{i=0}^{N}\frac{q^i}{q^N}. \tag{4.3}$$

Following the same reasoning for a generic K>N, we obtain the following formula:

$$P_{ns}(K,N) = \frac{1}{q^{N(K-N)}}\prod_{j=1}^{N}(1-\frac{1}{q^j})\sum_{r_1=0}^{N}q^{r_1}\sum_{r_2=r_1}^{N}q^{r_2}\cdots\sum_{r_{K-N}=r_{(K-N-1)}}^{N}q^{r_{K-N}}. \tag{4.4}$$

Note that there are K-N summatories in the formula. However, this formula can be reduced considering the q-binomial coefficients, also called Gauss Coefficients. The q-binomial of two non-negative integers m and n is defined as:

$$\begin{bmatrix} m \\ n \end{bmatrix}_q = \frac{(q^m-1)(q^{m-1}-1)...(q^{m-n+1}-1)}{(q^n-1)(q^{n-1}-1)...(q-1)}. \tag{4.5}$$

Note that, if $n = 0$, the q-binomial has value 1 by definition, while, if $q = 1$, the Gauss Coefficient becomes the well known binomial coefficients. Although Gauss Coefficients appear as rational functions, they are in fact polynomial, since the denominator is always a factor of the numerator. It is not surprising that Gauss Coefficients appear in eq. (4.4), since, among others, a Gauss Coefficient counts the number $V\{m,n;q\}$ of different n-dimensional vector subspaces of an m-dimensional vector space over $GF(q)$.

Using the Gauss Coefficient properties eq. (4.6) and eq. (4.7):

$$\sum_{i=0}^{m-1} q^i = \begin{bmatrix} m \\ 1 \end{bmatrix}_q \tag{4.6}$$

$$\begin{bmatrix} m \\ n \end{bmatrix}_q = q^n \begin{bmatrix} m-1 \\ n \end{bmatrix}_q + \begin{bmatrix} m-1 \\ n-1 \end{bmatrix}_q \tag{4.7}$$

We may show for K-N=2 that

$$\begin{bmatrix} K \\ K-N \end{bmatrix}_q = \begin{bmatrix} N+2 \\ 2 \end{bmatrix}_q = \sum_{r=0}^{N} q^r \sum_{s=r}^{N} q^s \tag{4.8}$$

Now, it can be easily shown using these recursions that the embedded summatories are equal to $\begin{bmatrix} K \\ K-N \end{bmatrix}_q$, and grouping terms:

$$P_{ns}(K,N) = \begin{cases} 0 & \text{if } K < N \\ \prod_{j=1}^{N} (1 - \frac{1}{q^j}) \frac{\begin{bmatrix} K \\ K-N \end{bmatrix}_q}{q^{N(K-N)}} & \text{if } K \geq N, \end{cases} \tag{4.9}$$

By applying eq. (4.5), we can reduce eq. (4.9) to:

$$P_{ns}(K,N) = \begin{cases} 0 & \text{if } K < N \\ \prod_{j=0}^{N-1} (1 - \frac{1}{q^{K-j}}) & \text{if } K \geq N. \end{cases} \tag{4.10}$$

In eq. (4.10), $P_{ns}(K,N)$ represents the cumulative distribution function of the probability of receiving N linearly independent packets, given the transmission of $K \geq N$ packets under random linear network coding. The probability density function can then be computed as $p_{ns}(K,N) = P_{ns}(K,N) - P_{ns}(K-1,N)$, and the average number of packets to be sent in order to decode the N original ones is:

$$E[K] = \sum_{k=N}^{\infty} k \cdot p_{ns}(k,N). \tag{4.11}$$

As a further point, in a real implementation, the zero vector would be explicitly excluded from the extraction urn. When accounting for this aspect in the model, eq. (4.2) becomes:

$$P_{ns}(K,N) = \prod_{j=0}^{N-1} (1 - \frac{q^j - 1}{q^N - 1}) = (\frac{q^N}{q^N - 1})^N \prod_{j=1}^{N} (1 - \frac{1}{q^j}), \tag{4.12}$$

whereas eq. (4.4) results in:

$$P_{ns}(K,N) = \frac{q^{N^2}}{(q^N-1)^K} \prod_{j=1}^{N}(1-\frac{1}{q^j}) \\ \sum_{r_1=0}^{N}(q^{r_1}-1)\sum_{r_2=r_1}^{N}(q^{r_2}-1)\cdots\sum_{r_{K-N}=r_{(K-N-1)}}^{N}(q^{r_{K-N}}-1). \tag{4.13}$$

Expressing eq. (4.13) in terms of q-binomial coefficients, with

$$P_{ns}^0 = \frac{q^{N^2}}{(q^N-1)^K}\prod_{j=1}^{N}(1-\frac{1}{q^j}) \tag{4.14}$$

results in:

$$P_{ns}(K,N) = \begin{cases} 0 & \text{if } K < N \\ P_{ns}^0 \cdot \left(\begin{bmatrix} K \\ K-N \end{bmatrix}_q + \sum_{n=1}^{K-N}(-1)^n \binom{K}{n} \begin{bmatrix} K-n \\ K-N-n \end{bmatrix}_q \right) & \text{if } K \geq N. \end{cases} \tag{4.15}$$

As a final remark, we stress that the formulation above only accounts for the decoding probability due to the actual random coding, in terms of excess packets and field size. Packet losses and network topology also impact on the number of packets required for a correct transmission, but they are independent of how random vectors are chosen. For example, eq. (4.16) accounts for the probability of decoding N packets given that a node receives K coded packets in a channel, where $p_{channel}(n)$ is the probability that n packets are correctly received.

$$p_{dec} = \sum_{j=N}^{K}\binom{K}{j}p^j(1-p)^{K-j}\cdot p_{ns}(j,N) \tag{4.16}$$

4.3 Numerical Results

As discussed above, our analysis concerns the decoding probability as a function of q and S, and is thus limited to the linear independence of the random vectors employed for the encoding of received packets. Losses due to channel errors represent an additional issue, that is not the focus of our study. Therefore, we will thus assume no packet losses in the following. Figure 4.1 shows $P_{ns}(K,N)$ as a function of the number of excess packets, S, for N=100 and $q \in \{2,4,16,32\}$. Our analysis, represented by continuous and dashed lines, perfectly matches simulation results, portrayed as points, that are obtained via

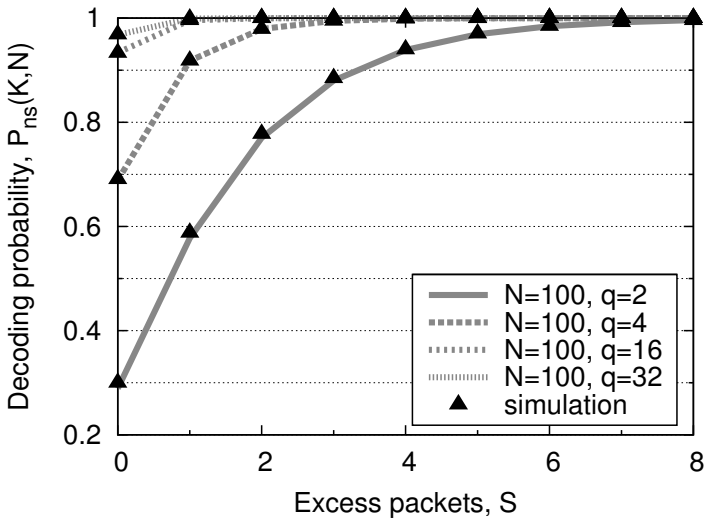


Figure 4.1: P_{ns} versus S, for N=100

an actual extraction of random vectors, and are averaged over 10^5 runs: differences between analytical and simulative values are in the order of 0.1%.

We can also observe that larger values of the field size q allow to reach higher decoding probabilities with a same number of excess packets, or, conversely, less excess packets are required to reach a high decoding probability. However, increasing the field size only pays out for low values of q, since considering very large field sizes induces greater computational complexity, but no real advantage in terms of decoding probability.

On the other hand, the number of packets N has a negligible impact on the results, as depicted in Figure 4.2. There, the number of packets is N=5, but the results are virtually identical to those obtained for the case of N=100. Again, the analysis perfectly matches simulation.

Figure 4.3 shows the average number of excess packets $E[K]-N$ required to decode the N original packets. The plot portrays the outcome of our analytical formulation in eq. (4.15), the upper bound in (89) described in eq. (4.1), as well as the results obtained from simulation. Once more, our analysis provides a perfect matching with the simulation results.

Moreover, the exact formulation shows that the average number of excess packets required for the decoding is noticeably lower than that indicated by the upper bound.

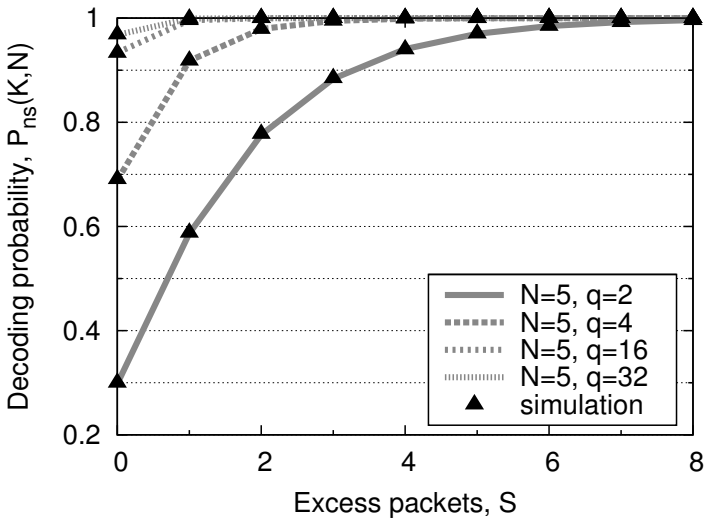


Figure 4.2: P_{ns} versus S, for N=5

Indeed, the exact solution demonstrates that, if a field size of q equal to 3 or 4 is considered, just one extra packet is sufficient for having N linearly independent packets. Such result also holds when considering large blocks of coded packets (i.e., high values of N), a situation that the upper bound cannot reproduce and that was only discussed via simulation in (89).

As a final remark, we note that numerical results on the decoding probability in presence of the zero vector, as in eq. (4.10), returned values similar to those shown for eq. (4.15), unless very small values of q and N are selected.

4.4 Conclusions

We have computed the exact probability that a receiver obtains N linear independent packets over $K \geq N$ received packets under random linear network coding over a Galois Field of size q. The derivation makes use of an urn model, and employs Gauss Coefficients to achieve a simple formulation.

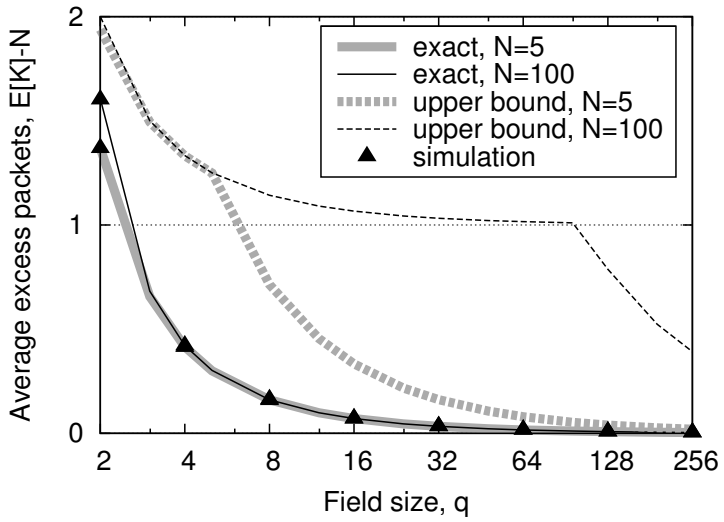


Figure 4.3: Mean excess packets to decode versus q

5

Planning Roadside Infrastructure for Information Dissemination in Intelligent Transportation Systems

5.1 Introduction

Wireless communications for intelligent transportation systems (ITS) are intended for the support of traffic safety and efficiency, as well as of value-added services, such as infotainment and commercial applications. The main components of the ITS architecture are roadside infrastructures and vehicles, and two main communication paradigms are foreseen, namely vehicle-to-vehicle (V2V) and vehicles-to-infrastructure (V2I). As a consequence, most of the research efforts so far have focused on the development of protocols and applications suitable for an ad hoc network composed of vehicles (the so-called VANET) and infrastructure nodes.

In this chapter, we deal with information dissemination to passing vehicles, tackling the specific issue of deploying an intelligent transportation infrastructure that efficiently achieves the dissemination goal. More specifically, we consider a system that has to support information dissemination or lookup and retrieval, for such purposes as parking lot availability, transportation timetables, pollution data collection. Then we pose the following question: assuming that an area, with an arbitrary road topology, must

be equipped with a limited number k of infrastructured nodes (e.g., IEEE 802.11 access points), what is the best deployment strategy to maximize the dissemination of information?

In principle, an information dissemination system could leverage both V2V and V2I communications: when only few road-side units can be deployed, V2V communications enable data sharing thus increasing the throughput perceived by the users while downloading a content. However, to deploy an infrastructure that optimally provides its services to the users, while maximizing the benefit coming from V2V data sharing, would require detailed knowledge of the vehicle mobility patterns and of the communication opportunities between vehicles. Due to both privacy issues and the complexity of collecting data, it is unlikely that such information will be available for generic user vehicles. Thus, in this work we restrict our attention to the problem of optimally positioning the infrastructure nodes, without tackling V2V communications.

We refer to the infrastructured nodes as *Dissemination Points* (DPs), and, as a first step to the solution of our problem, we show that road intersections are preferred locations to place DPs. Then, we address two different cases. Firstly, we assume that the information is just a small, self-contained item. A vehicle will receive the information item if it gets in contact with a DP at least once. Under this assumption, we are interested in placing the DPs at k of the possible intersections so as to maximize the number of vehicles that enter a DP coverage area at least once; we therefore model our problem as a Maximum Coverage Problem (MCP). Secondly, we consider the case in which vehicle-to-DP contact times have an impact on the dissemination process. In this case, we give a different formulation for our problem, which aims at favoring both the number of contacted vehicles as well as the contact times. Both versions of the problem, however, are NP-hard, thus we propose heuristic algorithms for their solution. Note that other performance metrics and, thus, optimization objectives could be also considered (e.g., minimizing the information dissemination time), but, again, it would require further assumptions on the knowledge that is possible to acquire (e.g., on content size, per-vehicle link data rate, etc.).

The performance of our heuristics is evaluated by considering a real-world urban environment and realistic vehicular traces. More specifically, we use traces of vehicular mobility in the canton of Zurich that have a duration of one hour and a half (82). We point out, however, that in presence of very long traces, our models and solutions

should be applied to rush-hour representations of the vehicular traffic, as typically done in system planning.

5.2 Related Work

The Set Covering Problem (SCP) is a fundamental problem in complexity theory and was shown to be NP-Complete in 1972. SCP has been used to model many application problems such as routing, service planning, manufacturing, scheduling, etc. There are several surveys, E. Balas (90), Ceria et al (91), listing problems formulated and solved as SCP. Paschos (92) surveys approximation algorithms for covering and packing problems, including weighted and unweighted algorithms. Grossman et al (93) compare nine different algorithms for the SCP, six of which (Gr: Greedy; NoLP: generalization of Gavril's algorithm; Thres: Hochbaum's algorithm; Alt-Gr: alternating greedy and SortLP and T-Gr:) are deterministic and three (R-Gr: randomized greedy; RR: randomized rounding; NN: neural network) are randomized. Other authors such as Caprara (94) also survey algorithms for the SCP giving exhaustive lists of heuristic and exact approaches. Finally, Pisinger (95) gives an overview of exact solutions for Knapsack problems.

Wireless Access Point or Base Station placement is a well known research topic, however most of the works that have addressed this problem so far consider a continuous infrastructure radio coverage. Wright (96) proposes a variant of the Nelder-Mead "simplex" method for finding optimal base station placement. Whitaker et al. (97) investigate macro-cell network planning in cellular networks and analyze the effects of cell density on the infrastructure cost of the network and the effects of increasing infrastructure expenditure on service coverage. Other similar works use set covering problems for network planning: Tutschku (98) introduces the Set Cover Base Station Positioning Algorithm that is based on a greedy heuristic for solving the Maximal Coverage Location Problem in radio networks, while Amaldi et al. (99) study WLAN positioning taking into account IEEE802.11 access mechanism. None of the former works however consider wireless networks with intermittent coverage, where base stations are planned taking into account non-continuous coverage for mobile users. Recently, Lochert et al. (76) have tackled the problem of sparse Road Side Units placement, formulating it as an optimization problem solvable with genetic algorithms. However, the goal of

the deployment is in that case the aggregation of data on vehicular traffic conditions, and not the dissemination of information: the diverse target leads to a different problem formulation and, thus, solution. An interesting work is also presented in (100), where Chaintreau et al. focus on opportunistic communications between nodes that move between classes; classes may represent locations or states. The authors show that gossiping through opportunistic contacts (e.g., between taxi cabs) lead to efficient information update, and exploit their findings for base station deployment.

Our architecture assumes the InfoStation model (101), where small islands of coverage provide low-cost information services for mobile users. In the area of InfoStation models, the objective of most of the works that have appeared in the literature has been routing and reliability of information delivery, but not optimal AP placement. As an example, Sollazzo et al. (102) propose TACO-DTN, a content dissemination system that uses the InfoStation model, and study the routing governance and management of the AP resources when the AP disseminates information. Cohen et al. (103) consider the case where a set of Information Dissemination Devices disseminate information to passing mobile nodes. The authors use the Knapsack formulation to decide which messages should be broadcast by every dissemination device.

In the context of sensor networks, several studies (104, 105, 106) have considered the problem of deploying multiple sinks in presence of stationary nodes. The goal of these works, however, is to minimize energy consumption, while guaranteeing that sensors can access at least one sink through either single- or multi-hop communications. In the case of mobile sensors (107), the problem still differs significantly form ours, not only because of the different objectives, but also because of the different nodes mobility. Sensors mobility is typically represented by random models or follow pre-defined trajectories, also it is characterized by a much lower speed than vehicular mobility; it follows that, unlike our case, the nodes movement can be predicted and exploited for sink deployment.

Finally, Maeda et al. (108) have proposed a technique to derive the movement of pedestrians in an urban scenario, starting from the street layout and observations of density at intersections. A similar technique could be exploited to derive (otherwise unavailable) information on the exact mobility of cars from measurements of congestion at crossroads, since we proved how knowledge of the former is much more useful than

knowledge of the latter in order to achieve an optimal DPs deployment for information dissemination.

To summarize, our study differs from previous work in that it deals with optimal deployment of dissemination points when full coverage of the network area is not required and nodes have intermittent connectivity with the network infrastructure. Also, it addresses the case of generic user vehicles and, as is conceivable, it assumes that limited information is available on vehicle mobility as well as on their communication capabilities.

5.3 System Scenario and Goals

We consider a urban road topology of area size equal to A, including N intersections. We assume that each DP has a dissemination range equal to R. Such a dissemination range may map into the DP's transmission range, or into its service range if dissemination can be performed through multihop communication. Also, we denote by V the number of vehicles that transit over the area A during a given time period, hereinafter called observation period.

Our goal is to deploy k DPs so as to maximize either the number of vehicles, among the possible V, served (i.e., covered) by the DPs, or to favor both the number of covered vehicles and the connection time between vehicles and DPs. Note that this significantly differs from other coverage problems, since

- the DPs deployed in the area do not have to necessarily form a connected network, or provide a continuous coverage of the road topology; also, energy saving is not one of our goals. These are major differences with respect to previous work on maximum graph coverage (109) as well as on cellular and sensor wireless networks (see e.g., (97, 104, 105, 106));

- vehicles directly access the DPs, in addition their movement obeys to traffic regulations and is constrained by the road topology; as a consequence, the scenario differs from the one studied in (110) for the deployment of Internet access points in static networks, or from mobile sensor networks (107);

- vehicles may cross several intersections, thus they may be covered (i.e., served) by more than one DP. When contact time is taken into account, this aspect makes existing generalizations of the MCP unsuitable to our problem.

In the following, we deal with the problem of planning vehicular networks for information dissemination, taking into account the above issues and the peculiarities of these systems.

5.4 Selecting the Location Type

The evaluation of *where* on a road to deploy the DPs is an important first step in designing an efficient dissemination system for vehicular environments. Nominally, the position of a DP over a single road segment can span anywhere between adjacent intersections: thus, the problem basically lies in deciding whether a DP should be located midway through the road segment, or closer to the intersections bounding it.

To this end, we simulate a realistic vehicular mobility over a simple road topology, and measure the potential for information dissemination of an individual DP, deployed at first in the intermediate point of a road segment, and then at an intersection ending the same street. The movement of vehicles is simulated with VanetMobiSim (111), employing the IDM-LC model, which reproduces car-to-car interactions, stopping, braking and acceleration phenomena in presence of traffic lights at road junctions, and overtaking, as observed in real world (112).

In particular, we considered different vehicular lane densities, ranging between 5 and 20 vehicles/km, which map to low and dense traffic conditions, respectively. The potential for dissemination is evaluated in terms of number of concurrent vehicle-to-DP contacts and of time spent by each vehicle within the dissemination range R of the DP: a higher number of vehicles, as well as longer permanence times, are indicative of a higher potential for information dissemination, as more users can receive larger portions of the content provided by the DP.

Fig. 5.1 depicts the Cumulative Density Function (CDF) of such two metrics, when the DP is positioned along the road or at the intersection, with varying vehicular densities. It can be observed that the car density has a negligible impact on the time that vehicles spend within DP's dissemination range, while it strongly impacts the number of vehicles in that same area. In both cases, however, deploying the DP at the

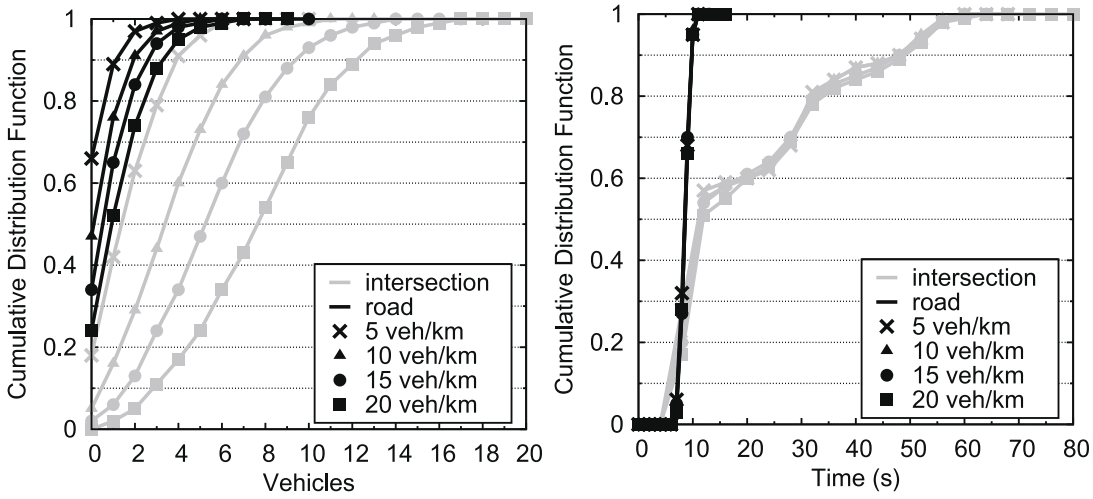


Figure 5.1: CDF of the number of vehicles within range of the DP (left) and of the time spent by vehicles within range of the DP (right), with $R = 50$ m and for different vehicle densities

intersection leads to better results, since more vehicles travel through the dissemination area, spending there a longer time.

We also analyzed the effect that different DP ranges have on the dissemination performance. Fig. 5.2 portrays the same metrics studied before, for several values of R. The dissemination range significantly affects both CDFs, with larger ranges clearly providing better performance. In any case, deploying the DP at the intersection yields again more favorable properties than positioning it along the road, for any value of R.

According to these results, intersections prove to be much better locations than road segments for the deployment of DPs, in terms of information dissemination potential. Thus, in the reminder of the paper, we will focus on the problem of DPs deployment at intersections of the road topology.

5.5 Deployment Algorithms

As stated before, we consider two cases, accounting for (i) only the number of vehicles that get in contact with DPs, and (ii) both the number of served vehicles and the vehicle-to-DP contact times.

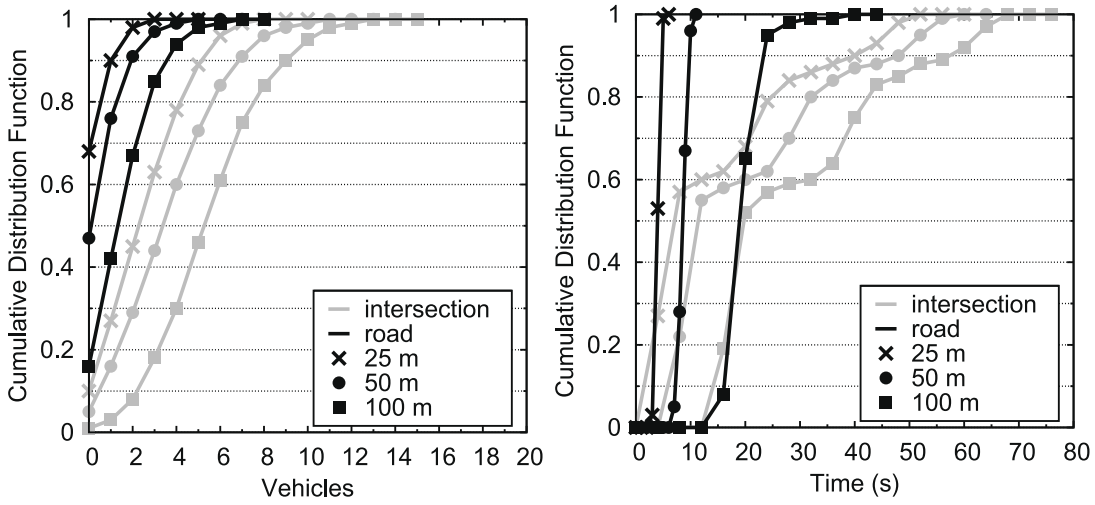


Figure 5.2: CDF of the number of vehicles within range of the DP (left) and of the time spent by vehicles within range of the DP (right), with a vehicular lane density of 10 vehicles/km and for different DP dissemination ranges

5.5.1 Maximizing contacts

Our goal is to maximize the number of vehicles covered by k DPs. Based on the above results, we constrain ourselves to considering only the N intersections located in the road topology as possible locations for a DP. In particular, by analyzing the vehicular mobility in the selected area, we define an $N \times V$ matrix $\mathbf{P}$ whose generic element is given by

$$\mathbf{P}_{ij} = \begin{cases} 1 & \text{if vehicle } j \text{ crosses intersection } i \\ & \text{during the observation period} \\ 0 & \text{otherwise} \end{cases} \tag{5.1}$$

It is worth pointing out that the use of matrix $\mathbf{P}$ requires that the identity of each vehicle be known so that it can be tracked across all intersections. (In Section 5.5.1, we will relax this assumption and present an approach where the identity need not be recorded.)

We model the problem as a Maximum Coverage Problem (MCP), which can be formulated as follows. We are given a collection of sets $\mathcal{S} = \{S_1, S_2, \ldots, S_N\}$, where each set S_i is a subset of a given ground set $X = \{x_1, \ldots, x_V\}$. The goal is to pick k sets from $\mathcal{S}$ to maximize the cardinality of their union.

To better understand the correspondence with our problem, consider that the elements in X are the vehicles that transit over the considered road topology during the

observation period. Also, for $i = 1, \ldots, N$ we have

$$S_i = \{x_j \in X,\ j = 1, \ldots, V : \mathbf{P}_{ij} = 1\} \tag{5.2}$$

i.e., S_i includes all vehicles that cross intersection i at least once over the observation period. Thus, by solving the above problem, we obtain the set of k intersections where a DP should be placed so as to maximize the number of covered vehicles.

Unfortunately, the MCP problem is NP-hard; however, it is well known that the greedy heuristic achieves an approximation factor of $1 - (1 - \frac{1}{m})^m$, where m is the maximum cardinality of the sets in the optimization domain (113). We report the greedy heuristic below.

The greedy algorithm

The greedy heuristic (hereinafter also called MCP-g) picks at each step a set (i.e., an intersection) maximizing the weight of the uncovered elements.

Let us introduce an auxiliary set G. Let $G \subseteq \mathcal{S}$ be a collection of sets and W_i $(i = 1, \ldots, N)$ be the number of elements covered by S_i, but not covered by any set in G. The steps of the greedy heuristic are reported in Algorithm 1.

Algorithm 1 The MCP-g heuristic

Require: k, $\mathbf{P}$, $\mathcal{S}$

1: $G \leftarrow \emptyset$, $C \leftarrow 0$, $U \leftarrow \mathcal{S}$
2: $W_i = \sum_{j=1}^{V} \mathbf{P}_{ij},\ i = 1, \ldots, N$
3: **repeat**
4: Select $S_i \in U$ that maximizes W_i
5: $G \leftarrow G \cup S_i$
6: $C \leftarrow C + 1$
7: $U \leftarrow U \setminus S_i$
8: $W_i = \sum_{\substack{j=1 \\ j:x_j \notin G}}^{V} \mathbf{P}_{ij},\ i = 1, \ldots, N$
9: **until** $C = k$ or $U = \emptyset$

Note that, although such algorithm provides a very good approximation of the optimal solution, it requires:

i) global knowledge of the road topology and network system,

ii) the identity of the vehicles which have crossed the N intersections during the observation period.

Below, we propose *i)* a hierarchical algorithm which reduces the computational complexity by applying the *divide et impera* approach, and *ii)* a different problem formulation where the knowledge of the vehicles identity is not needed.

The subzone algorithm

We superimpose an overlay grid with cells of arbitrary, equal size on our road topology. We name a cell as *subzone* and denote the number of subzones by $B = 2^L$ (with $L \in \mathbb{N}_1$). We define a hierarchical structure consisting of $L + 1$ levels, such that, at the generic level l ($l = 0, \ldots, L$), the unit area includes 2^{L-l} subzones.

We start by solving the maximum coverage problem in each subzone (i.e., $l = 0$), and we find the optimum location of k_0 DPs in every overlay grid. Then, at each step $l \geq 1$, we divide the area of the grid into 2^{L-l} subzones, each twice the size of a single subzone at the previous step, and we select k_l intersections among the ones that were chosen at step $l - 1$. We repeat the procedure till the subzone area coincides with the area of the overlay grid (i.e., $l = L$).

The subzone heuristic, hereinafter also called MCP-sz, is reported in Algorithm 2.

Algorithm 2 The MCP-sz heuristic

Require: k, $\mathbf{P}$, $\mathcal{S}$, $1 < B = 2^L$

$\mathcal{S}' \leftarrow \mathcal{S}$
for $l = 0$ to L **do**
  Divide the road topology into 2^{L-l} cells of equal size
  for $m = 1$ to 2^{L-l} **do**
    Solve the MCP in the m-th subzone, by taking $\mathcal{S}'$ as input set and k_l as the number of DPs to deploy
    Remove from $\mathcal{S}'$ the unselected intersections
    $m \leftarrow m + 1$
  end for
  $l \leftarrow l + 1$
end for

Note that the value of k_l can be set so as to limit the number of intersections selected within each subzone at step l ($l = 0, \ldots, L$). As an example, for $k \ll N$, we found that the algorithm can be efficiently run by fixing $k_l = k$, $\forall l$. For larger values of k, instead, setting $k_l = \lceil \frac{k}{2^{L-l}} \rceil + 2^{L-l} - 1$ allows the selection of at least $\frac{k}{2^{L-l}}$ per subzone, i.e., k intersections in the whole area, plus some extra intersections per subzone ($2^{L-l} - 1$). The benefit of such redundancy is twofold: it allows us to better approximate a centralized solution, and its impact is limited since the number of extra intersections reduces exponentially at each step of the procedure till it reaches 0 at the last round (i.e., $l = L$).

As a last remark, the value of B can be determined so as to limit the number of candidate intersections that are selected at each round (hierarchical level) of the procedure. In particular, given k_0, the number of intersections selected in the first round ($l = 0$) must be less than or equal to the number of existing intersections, i.e.,

$$Bk_0 \leq N \tag{5.3}$$

Since $B = 2^L$, from (5.3), it is possible to derive a value for L and, thus, for the number of levels that avoids useless iterations, i.e., to consider too fine grids which do not yield any selection of intersections.

Unknown vehicles identity

Unlike the previous case, we now assume that the vehicles identity is not recorded and the only available information is the number of different vehicles that have crossed each of the N intersections during the observation period. Thus, our objective becomes the maximization of the total number of service opportunities provided by k DPs.

To this end, let ν_i, $i = 1, \ldots, N$, be the total number of vehicles that have crossed intersection i during the observation period, i.e.,

$$\nu_i = \sum_{j=1}^{V} \mathbf{P}_{ij} \qquad i = 1, \ldots, N \tag{5.4}$$

We then model the problem as a 0-1 Knapsack Problem (KP), which is defined as follows (95). We are given a bag and a set of N items $\mathcal{I} = \{I_1, \ldots, I_N\}$. Each item $I_i \in \mathcal{I}$ has a non-negative value and a non-negative weight, and the maximum weight that we can carry in the bag is equal to k. The objective is to select a subset of items

$\mathcal{J}' \subseteq \mathcal{J}$ whose weight does not exceed k and that maximizes the overall value of the bag. Each item must only be selected once.

To better understand the correspondence with our problem, consider that the elements in I are the intersections; each intersection i has a weight equal to 1 and a value equal to ν_i $(i = 1, \ldots, N)$. Thus, our problem can be formulated as,

$$\max \quad \sum_{i=1}^{N} \nu_i y_i \tag{5.5}$$

$$s.t. \quad \sum_{i=1}^{N} y_i \leq k; \quad y_i \in \{0, 1\} \ \forall i \tag{5.6}$$

The 0-1 KP is an NP-hard problem in general, however in our case, where all intersections have the same weight, it can be solved in polynomial time by simply sorting the intersections in decreasing order by their value, and selecting the first k intersections. We will refer to this algorithm as KP-P.

In Section 5.6.2, we present the deployment and coverage performance obtained by solving the MCP by brute force and through the greedy algorithm (MCP-g), compared against the cases where the hierarchical approach is used (MCP-sz) and where vehicles identity are not available (KP-P).

5.5.2 Maximum coverage and contact times

Here we address our second case, where k DPs have to be deployed at the road intersections so as to favor both the number of covered vehicles, as well as the time for which they are covered. To this end, let us define an $N \times V$ matrix $\mathbf{T}$ whose generic element, $\mathbf{T}_{ij}$ represents the total time that vehicle j would spend under the coverage of a DP if the DP were located at intersection i, i.e., the contact time between a vehicle j and a DP located at intersection i. Then, we formulate the following problem, which we name Maximum Coverage with Time Threshold Problem (MCTTP): given k DPs to be deployed, we aim at serving as many vehicles as possible, for (possibly) at least τ seconds each, i.e.,

$$\max \quad \sum_{j=1}^{V} \left[\min \left(\tau, \sum_{i=1}^{N} \mathbf{T}_{ij} y_i \right) \right] \tag{5.7}$$

$$s.t \quad \sum_{i=1}^{N} y_i \leq k; \quad y_i \in \{0, 1\} \ \forall i \tag{5.8}$$

Note that in (5.7) we place a DP at an intersection so as to maximize the number of vehicles that are covered, taking into account a vehicle's contact time up to a maximum value equal to τ: DPs that provide coverage for at least τ seconds to a given vehicle do not further contribute to the overall gain of covering such a vehicle. The constraint in (5.8) instead limits the number of DPs to k.

It can be easily verified that the MCP is a particular case of the above formulation, obtained by setting $\tau = 1$ and $\mathbf{T}_{ij} = \mathbf{P}_{ij}$. Hence, MCTTP is NP-hard and we propose the following heuristic for its solution.

A greedy approach

The greedy algorithm we propose to solve the MCTTP problem, denoted by MCTTP-g, picks an intersection at each step so as to maximize the provided coverage time, although only the contribution due to vehicles for which the threshold τ has not been reached is considered.

Let $G \subseteq \mathcal{S}$ be a collection of sets and let now W_i $(i = 1, \ldots, N)$ be the total contact time provided by intersection i, considering for each vehicle a contribution such that the vehicle's coverage time due to $G \cup S_i$ does not exceed the threshold τ. The greedy heuristic is reported in Algorithm 3.

Algorithm 3 The MCTTP-g heuristic

Require: k, $\mathbf{T}$, τ, $\mathcal{S}$

$G \leftarrow \emptyset$, $C \leftarrow 0$, $U \leftarrow \mathcal{S}$
$t_j = 0,\ j = 1, \ldots, V$
repeat
  $W_i = \sum_{j=1}^{V} \min(\tau - t_j, \mathbf{T}_{ij}),\ i = 1, \ldots, N$
  Select $S_i \in U$ that maximizes W_i
  $G \leftarrow G \cup S_i$
  $C \leftarrow C + 1$
  $U \leftarrow U \setminus S_i$
  $t_j = \min(\tau, t_j + \mathbf{T}_{ij}),\ j = 1, \ldots, V$
until $C = k$ or $U = \emptyset$

Again, we notice that the time-threshold heuristic requires knowledge of the global road topology and of the vehicles identity. Likewise for the MCP, we present a time-

subzone algorithm, which adopts the *divide et impera* approach and a 0-1 KP, for which knowledge of the vehicles' identity is not necessary.

The time-subzone algorithm

As done in Section 5.5.1, we divide the road topology in $B = 2^L$ cells, called subzones, and we apply the time-subzone heuristic (MCTTP-sz) whose steps are reported in Algorithm 4.

Algorithm 4 The MCTTP-sz heuristic

Require: k, $\mathbf{T}$, $\mathcal{S}$, $1 < B = 2^L$

1: $\mathcal{S}' \leftarrow \mathcal{S}$
2: Divide the road topology in B cells of equal size
3: **for** $l = 0$ to L **do**
4: **for** $m = 1$ to 2^{L-l} **do**
5: Solve the MCTTP in the m-th subzone, by taking $\mathcal{S}'$ as input set and k_l as the number of DPs to deploy
6: Remove from $\mathcal{S}'$ the unselected intersections
7: $m \leftarrow m + 1$
8: **end for**
9: $l \leftarrow l + 1$
10: Merge each pair of adjacent subzones so as to obtain 2^{L-l} subzones
11: **end for**

Unknown vehicles identity

When the vehicles' identities are not available, the only information we have is the total time that all vehicles would spend under the coverage of a DP if it were located at intersection i, i.e.,

$$T_i = \sum_{j=1}^{V} \mathbf{T}_{ij} \qquad i = 1, \ldots, N \tag{5.9}$$

Thus, in this case we want to maximize the total contact (service) time offered to the vehicles, when k DPs are deployed. Again, the problem can be formulated as a 0-1 KP. We are given a set of N intersections (items) $\mathcal{I} = \{I_1, \ldots, I_N\}$; each intersection has a value T_i and unitary weight, and the maximum number of selected

intersections (maximum weight) must be equal to k. The objective is to select a subset of k intersections that maximizes the overall service time provided to the vehicles, i.e.,

$$\max \quad \sum_{i=1}^{N} T_i y_i \tag{5.10}$$

$$s.t. \quad \sum_{i=1}^{N} y_i \leq k; \quad y_i \in \{0,1\} \ \forall i \tag{5.11}$$

As already mentioned, the above problem can be solved in polynomial time by using the simple algorithm reported in Section 5.5.1. We refer to this solution, which requires the knowledge of the T_i coefficients ($i = 1, \ldots, N$), as KP-T.

The performance of the brute force solution of the MCTTP problem are presented in Section 5.6.3, together with those of its greedy (MCTTP-g), subzone (MCTTP-sz), and no-identity (KP-T) heuristics.

5.5.3 Computational complexity of the proposed algorithms

The computational complexity of both MCP and MCTTP is $O(VN^k)$: given N intersections, all possible combinations where the k DPs can be placed have to be considered and the weight of each intersection is computed over V vehicles. The cost of both greedy heuristics, MCP-g and MCTTP-g, is $O(kVN)$, since, for k times, we have to select the best choice among the candidate intersections (initially set to N), and again the selection is based on the weight computed over V elements.

As for the MCP-sz and MCTTP-sz algorithms, we apply MCP and MCTTP, respectively, within each subzone. Being N/B (on average) the number of intersections within each subzone, the computational complexity of these algorithms results to be $O(B \log B \times V(\frac{N}{B})^k)$.

Finally, the complexity of the algorithm to solve the 0-1 KP is $O(VN + N \log N)$, since we just have to consider each of the N intersections and sort the values to obtain the best k choices.

5.6 Performance Evaluation

We applied the algorithms presented in the previous sections to a real-world road topology, in presence of realistic vehicular mobility. The resulting DP deployments were then evaluated in terms of information dissemination capabilities.

Table 5.1: Road topologies parameters

	Zurich	Winterthur	Baden	Baar
Intersections	83	43	38	46
Vehicles	70537	13578	11632	9876

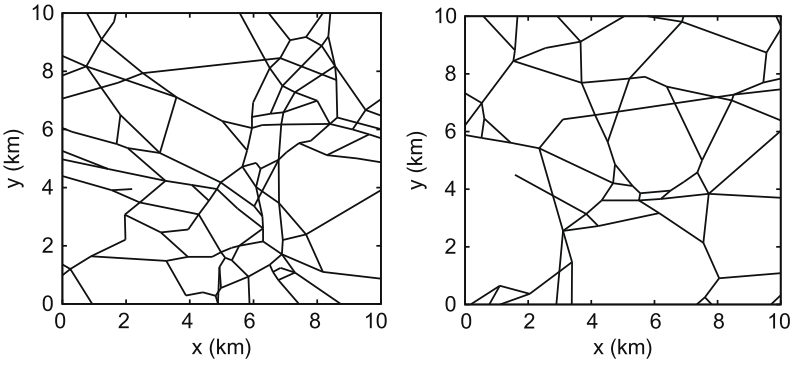
Here, we first introduce the evaluation scenario, and then we compare the results obtained with the different deployment algorithms maximizing contacts, as well as the results obtained with the different algorithms maximizing coverage and contact times.

5.6.1 Scenario

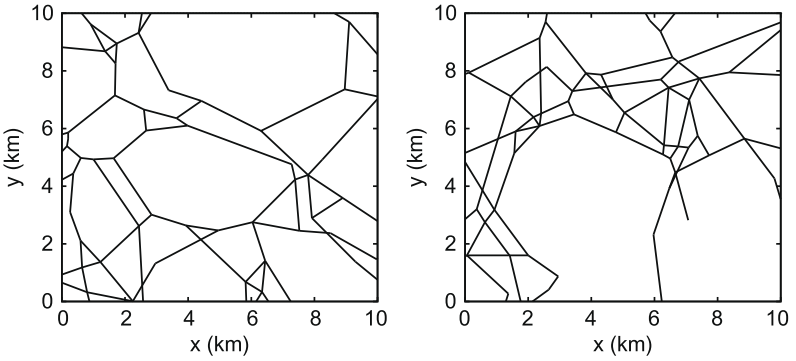
For our performance evaluation, we selected real-world road topologies from the canton of Zurich, in Switzerland. Realistic traces of the vehicular mobility in such region are available from the Simulation and Modeling Group at ETH Zurich (82). These traces describe the individual movement of cars through a queue-based model calibrated on real data (83): they thus provide a realistic representation of vehicular mobility at both microscopic and macroscopic levels.

We considered the four road topologies depicted in Fig. 5.3, representing 100 km^2 portions of the urban areas centered at the cities of Zurich, Winterthur, Baden, and Baar. For each topology, we extracted an hour and a half of vehicular mobility, in presence of average traffic density conditions. The number of road intersections in each scenario, and the amount of vehicles traveling within it during the observation time (after the filtering discussed below) are reported in Tab. 5.1.

In order to remove partial trips (i.e., vehicular movements starting or ending close to the border of the square area), we filtered the trace, by removing cars that traverse only three intersections or less, as well as those spending less than one minute in the considered region. Fig. 5.4 shows where the filtering thresholds fall with respect to the cumulative distribution functions of visited intersections and trip duration, for each scenario. The selected thresholds result in a low percentage of cars being removed from the traces of the scenarios characterized by a higher traffic density (Zurich and Winterthur), while the filtering is heavier on the traces of the more rural scenarios (Baden and Baar), where the conditions set above are harder to be met. However,



(a) Zurich (b) Winterthur



(c) Baden (d) Baar

Figure 5.3: Road topologies layouts

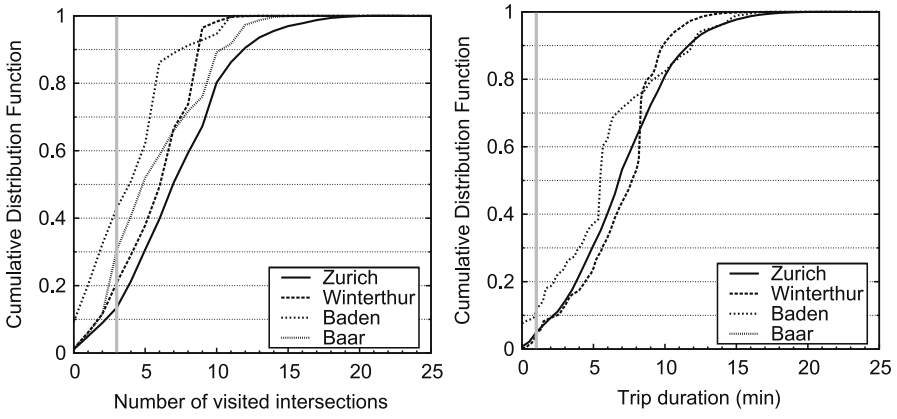
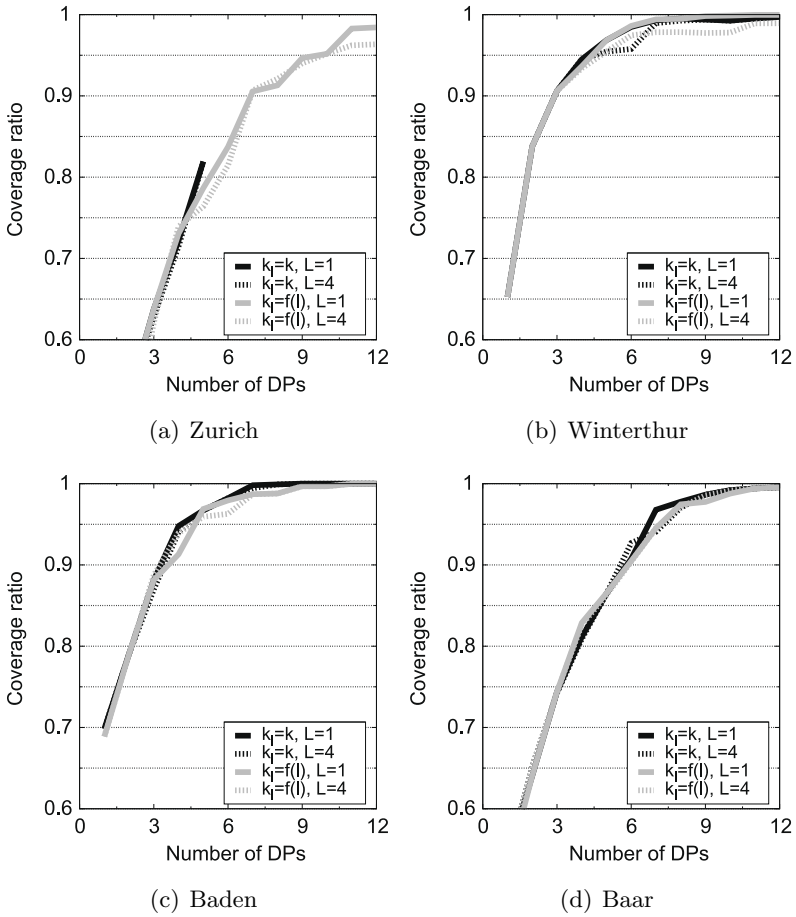


Figure 5.4: CDF of the number of intersections traversed (left) and of the trip duration (right) for all vehicles in the four traces

the resulting numbers, in Tab. 5.1, still guarantee the statistical validity of the tests conducted over all road topologies.

5.6.2 Maximizing contacts

In the scenario described above, we first run the deployment algorithms for contact maximization presented in Section 5.5.1. The metric we are interested in is the *coverage ratio*, i.e., the number of vehicles that experience at least one contact with a DP over the total number of vehicles in the scenario.



(a) Zurich (b) Winterthur

(c) Baden (d) Baar

Figure 5.5: MCP-sz calibration. In the label of each plot, $k_l = f(l)$ stands for $k_l = \lceil \frac{k}{2^{L-l}} \rceil + 2^{L-l} - 1$ (see Sec. 5.5.1)

The selected settings for the MCP-sz algorithm were L=4, k_l=k. This choice was the result of calibration tests run for MCP-sz on the different road scenarios, whose

outcome is shown in Fig. 5.5. There, we can notice how the impact of both the value of L and of the expression of k_l on the coverage ratio is very small, for the values of k (i.e., number of DPs deployed) that we are interested in. We thus picked $L = 4$, since it implies a stronger locality of decision and thus a reduced computational complexity, and $k_l = k$, as a simpler yet efficient choice with respect to more composite formulations.

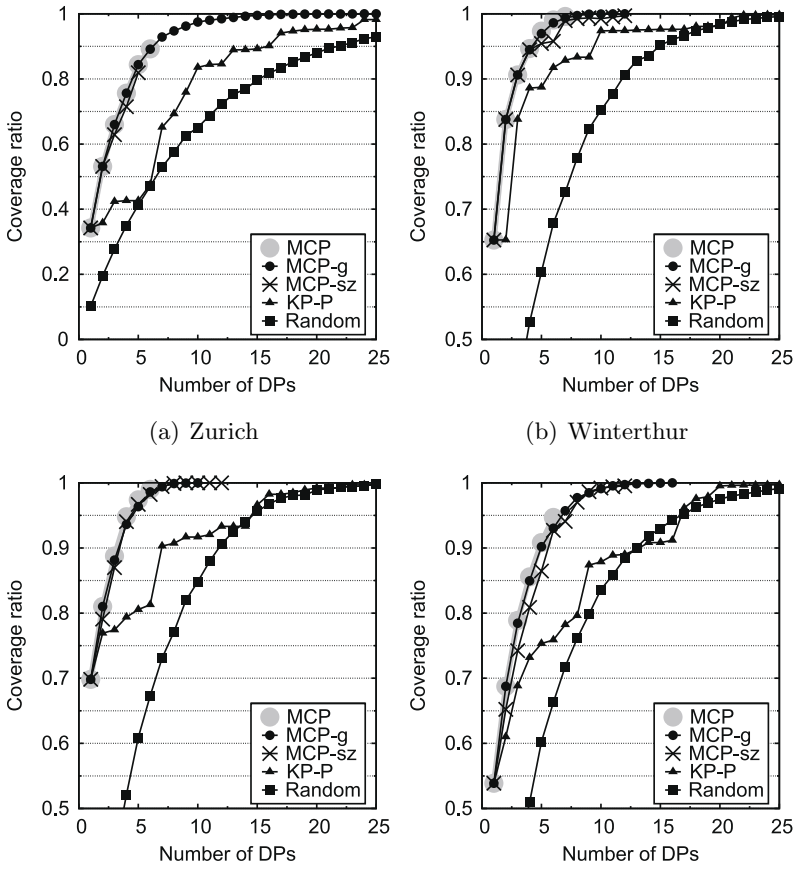
Also, in order to provide a lower-bound benchmark to the performance of the schemes, we tested the performance of a random deployment, that ignores the vehicular mobility information and whose outcomes result from averaging multiple tests over each road topology.

The coverage ratio achieved by the different contact maximization schemes is shown in Fig. 5.6, for each street layout. For each deployment algorithm, the ratio is recorded versus the number of allowed DPs k.

Three different behaviors can be distinguished in all the scenarios considered. The first is that of the random algorithm, which, lacking all information on the movement of vehicles, performs poorly: it needs a large number of DPs (typically more than 50% of intersections) to be deployed in order to provide one contact or more to each vehicle.

The second behavior is that of the KP-P scheme, which has only partial knowledge of the vehicular mobility, since it accounts for vehicular densities at intersections but neglects the mobility in between them. The KP-P algorithm performs better than the random one, although its absolute result still has wide margins for improvement. As a matter of fact, its CDFs grow faster than the random one, but do not reach again a coverage ratio equal to one until almost half of intersections are covered by one DP. Moreover, the progress in terms of covered vehicles is quite irregular as the number k of deployed DPs grows. This suggests that the KP-P scheme can at times deploy DPs at new intersections that do not improve the overall coverage.

The third behavior is that shown by the remaining algorithms: the brute force solution to the MCP, the greedy solution, and the subzone solution. The common point to these algorithms is that they all exploit full knowledge of the vehicles identity and mobility over the road topology. It is interesting to notice how both the greedy and the subzone schemes almost overlap with the optimal solution, and thus provide an excellent result in terms of information dissemination. Also, we stress that the difference with respect to the KP-P algorithm is extremely high, since the greedy and



(a) Zurich (b) Winterthur

(c) Baden (d) Baar

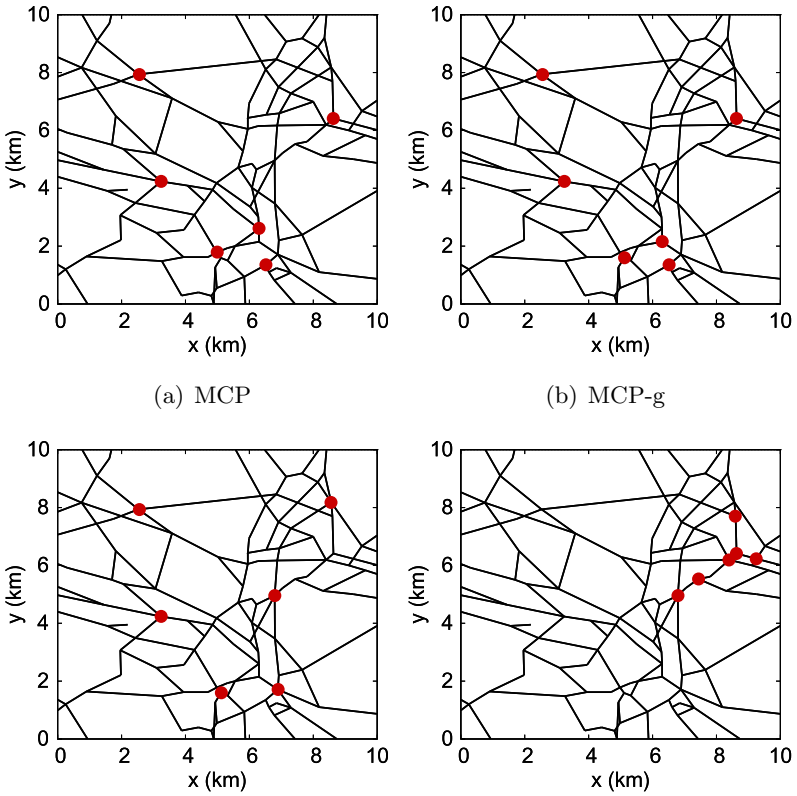
Figure 5.6: Ratio of vehicles experiencing at least one contact with a DP versus the number k of DPs deployed, for each road topology

subzone schemes cover 90% of vehicles with DPs covering between 5% and 10% of the available intersections, and 100% of vehicles with DPs at around 15% of intersections.

The variability in the percentages above is due to the different scenarios we consider. We also note that, although the relative performance of the algorithms stay the same for the different street layouts, the absolute coverage ratios change (note the different ranges in the y axes of the plots). More precisely, the higher complexity of the road topology in the Zurich area results in lower coverage ratios when just a few DPs are present, whereas a single DP is sufficient to already cover more than 50% of cars in the other three scenarios. Indeed, in a smaller urban center, most traffic tends to gather over one or two main roadways, and it is thus easier to cover by placing DPs at

strategic intersections. In larger metropolitan areas, instead, the vehicular flows split over a number of major routes, which makes it harder to cover them with a limited number of DPs.

Finally, we stress that the brute force solution to the MCP problem is computationally feasible only for low values of k, for which a 100% coverage cannot be usually reached. On the other hand, heuristics can be run even when DPs are deployed at a high percentage of the available intersections. This makes the fact that MCP-g and MCP-sz achieve performance similar to those of MCP an extremely important result, which allows to determine computationally-feasible quasi-optimal placements of DPs when a coverage close or equal to 100% is the goal of the deployment.



(a) MCP (b) MCP-g

(c) MCP-sz (d) KP-P

Figure 5.7: Zurich road topology: deployments of DPs obtained with different algorithms maximizing contacts, for $k = 6$

Further insight in the different behaviors is provided by Fig. 5.7. The figure shows

the actual positions of the DPs over the Zurich road topology, when $k = 6$, for the MCP, MCP-g, MCP-sz, and KP-P formulations. There, it can be observed how the greedy algorithm results in a solution that is nearly identical to the optimal one, whereas the subzone solution is less similar to the optimal, but still close to it. The reason is that the hierarchical approach trades the reduction in complexity for optimality, and can take suboptimal decisions during initial iterations. However, the final result is still very close to that obtained by solving the MCP by brute force. On the contrary, the deployment achieved by the KP-P algorithm is noticeably different, as DPs tend to be gathered in a same area, characterized by high vehicular traffic density. Since the selected intersections are close to each other, a high number of vehicles travels through several of the deployed DPs, so that most of the DPs have a very small impact on the coverage.

By summarizing the results, we can conclude that:

1. knowledge of vehicular trajectories is the discriminating factor in achieving an optimal deployment of DPs;

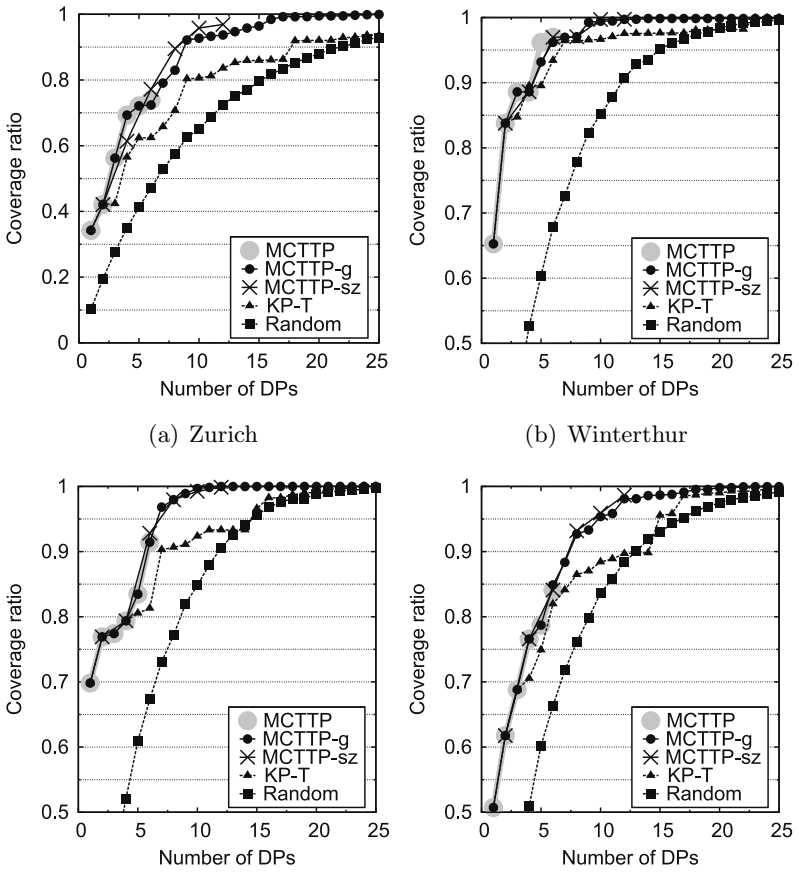
2. when exploiting such a knowledge, even a computationally feasible, hierarchical solution, such as the subzone algorithm can lead to near-optimal results in real-world road topologies of tens of km^2;

3. by exploiting these properties, it is possible to inform a high percentage of vehicles by deploying DPs at a small percentage of intersections.

5.6.3 Maximizing coverage and contact times

Taking into account the time dimension, we increase the complexity of the problem, by maximizing coverage and contact times between vehicles and DPs. Thus, in this case we are not only interested in the coverage ratio as a metric, but also in the *coverage time*, i.e., the amount of time that each vehicle spends within range of DPs during its trip in the considered scenario. Once more, a random deployment is employed to benchmark the performance of the algorithms we introduced in Section 5.5.2.

The coverage ratio achieved by such algorithms, in different road topologies and as the number of deployed DPs k varies, is depicted in the plots of Fig. 5.8, for a time threshold value $\tau = 30$ s. Exactly as observed in the previous section, also in this case



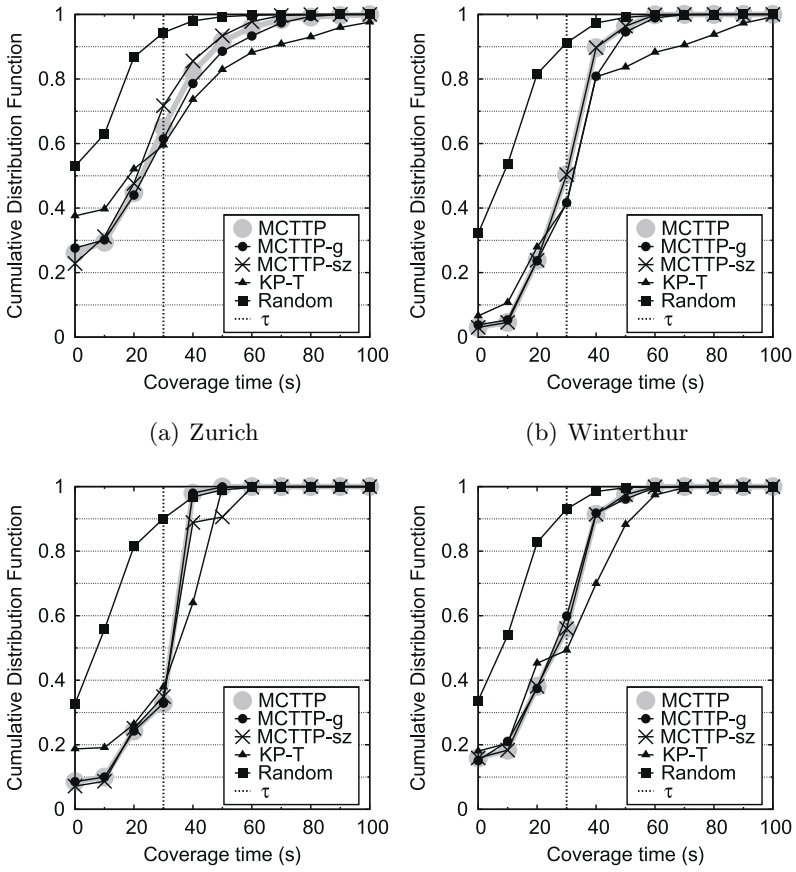
(a) Zurich (b) Winterthur

(c) Baden (d) Baar

Figure 5.8: Coverage ratio versus the number k of DPs deployed, for $\tau = 30$ s

the information on vehicular mobility plays a major role in favoring contacts among vehicles and DPs. As a matter of fact, the random solution performs poorly, while the KP-T algorithm provides a better coverage of the vehicles. The MCTTP, MCTTP-g, and MCTTP-sz solutions, leveraging their knowledge of cars trajectories, guarantee the highest coverage and tend to perform similarly. Such result is consistent through all scenarios, although the entity of the difference in the coverage ratio provided by the diverse deployment algorithms varies with the road topology considered: also in this case, a more complex road topology, such as that of Zurich, leads to more significant differences between the schemes that are mobility-aware and those that are not.

Fig. 5.9 reports instead the distribution of the coverage time, for $\tau = 30$ s (as remarked by the vertical threshold line in the plots), and in the specific case in which



(a) Zurich (b) Winterthur

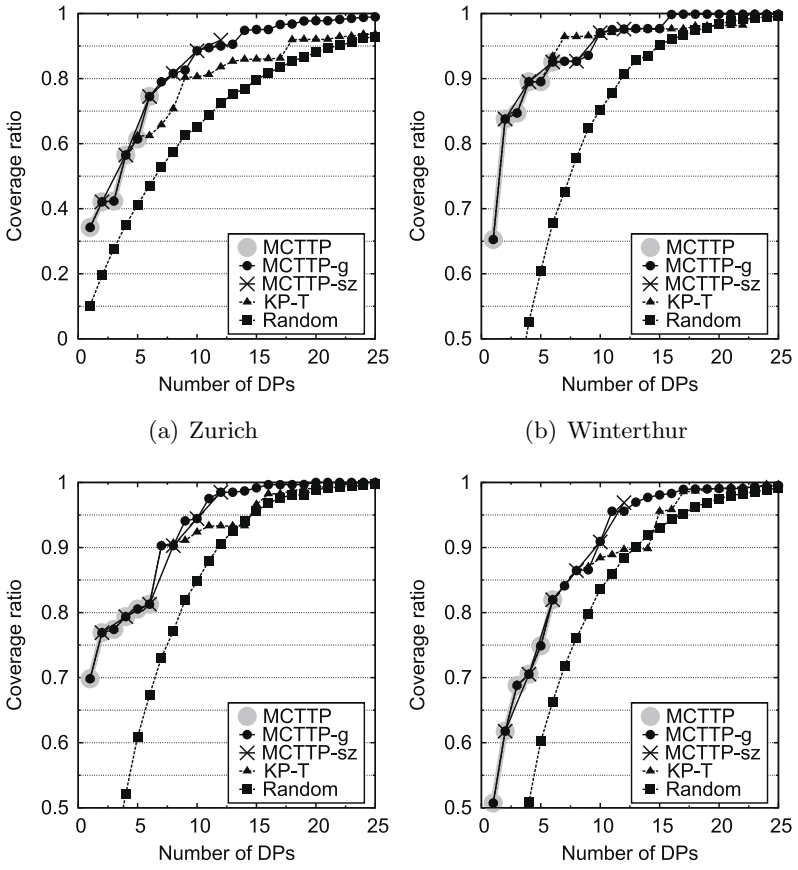
(c) Baden (d) Baar

Figure 5.9: Cumulative distribution function of the coverage time, for $\tau = 30$ s and k=6

$k = 6$ DPs are deployed over each road topology. The goal now is to maximize the time spent by vehicles under coverage of DPs, up to the threshold τ seconds. The common result in all road topologies is that random deployments lead to small coverage times, whereas the other schemes tend to behave similarly, although KP-T is characterized by a more skewed distribution than those of MCTTP, MCTTP-g, and MCTTP-sz. As a matter of fact, the deployments determined by KP-T at a time result in more vehicles with very low coverage times, and more vehicles with very high coverage times. Conversely, MCTTP, MCTTP-g, and MCTTP-sz lead to more balanced distributions, where many vehicles experience a coverage time around the threshold τ. Once more, these observations hold for all the scenarios considered.

When comparing the coverage times in Fig. 5.9 with the coverage ratios in Fig. 5.8,

we can notice that MCTTP, MCTTP-g, and MCTTP-sz provide very similar performance, which is superior to those achieved by the other schemes, for the considered settings $\tau = 30$ s and $k = 6$. Indeed, a random deployment of DPs induces both a lower number of vehicle-to-DP contacts and a shorter coverage time with respect to the solutions above. The KP-T solution leads to a performance comparable to those of MCTTP and relative heuristics in terms of coverage time, although with the skewness discussed before; however, this result is paid at a high coverage ratio cost.

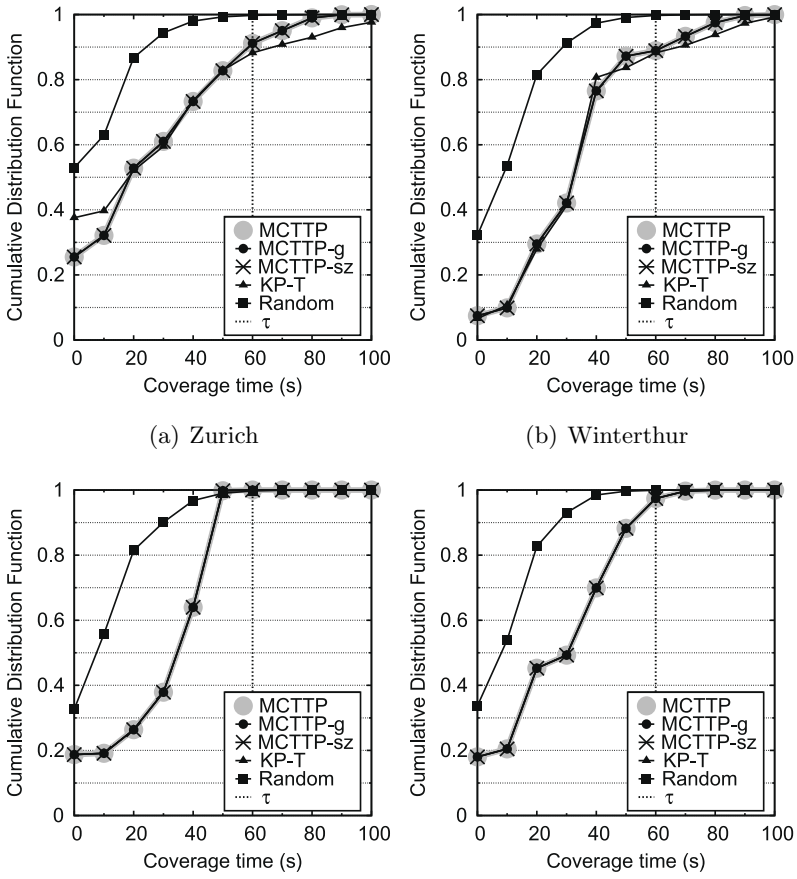


(a) Zurich (b) Winterthur

(c) Baden (d) Baar

Figure 5.10: Coverage ratio versus the number k of DPs deployed, for $\tau = 60$ s

When the value of the time threshold τ is increased, the constraint on the coverage time becomes stricter. Fig. 5.10 and Fig. 5.11 show, respectively, the coverage ratio (for varying k) and the CDF of the coverage time (for $k = 6$), when τ is set to 60 s. In the plot of Fig. 5.11, the threshold value τ is highlighted by the vertical dashed line.



(a) Zurich (b) Winterthur

(c) Baden (d) Baar

Figure 5.11: Cumulative distribution function of the coverage time, for $\tau = 60$ s and $k = 6$

We can note how such coverage time threshold is very hard to reach with the limited number of DPs we bind our deployment to: the percentage of vehicles covered for τ seconds when $k = 6$ is quite small, forcing the deployment of more DPs, or the increase of their dissemination range.

In any case, the random and KP-T schemes still perform worse than MCTTP and its greedy and subzone versions. In turn, the latter algorithms result in similar, but slightly reduced coverage ratios with respect to the case where smaller values of τ are considered, in an attempt to cover each vehicle for a longer time and match the threshold requirement.

The relationship of the τ-dependent schemes from τ is studied in Fig. 5.12, for

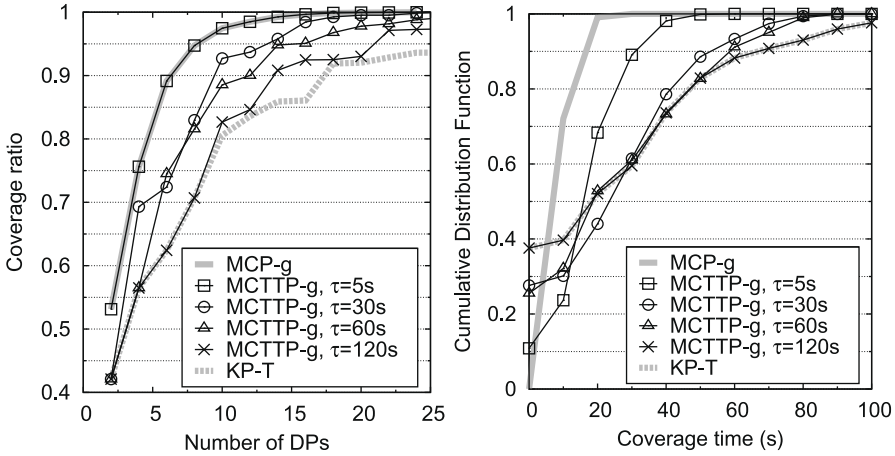


Figure 5.12: Zurich road topology: coverage ratio versus the number k of DPs (left) and coverage time CDF, for $k = 6$ (right)

the complex Zurich road topology. There, we focus on MCTTP-g, since the other algorithms showed similar behaviors, and evaluate it as τ ranges between 5 and 120 seconds. The coverage ratio, in the left plot of Fig. 5.12, shows how the MCTTP-g solution falls in between those obtained with an algorithm that maximizes vehicle-to-DP contacts, i.e., MCP-g, and with one that maximizes the overall coverage time, i.e., KP-T. In particular, for low values of τ, MCTTP-g tends to MCP-g, since the time constraint is easily satisfied (a contact with a single DP is often sufficient to reach the desired coverage time) and the algorithm can thus focus on maximizing the coverage. Instead, when τ is high, MCTTP-g tends to KP-T, since the desired coverage time is seldom reached, and thus the same vehicles keep on contributing to the optimization: the focus of the algorithm then shifts onto coverage times.

This is confirmed by the CDFs of the coverage time, on the right plot of Fig. 5.12, where the same behavior of the MCTTP-g algorithm is observed, as τ varies. It can be noted, however, how MCTTP-g with $\tau = 5$ s matches MCP-g in terms of coverage ratio, but outperforms it in terms of coverage time. Similarly, MCTTP-g with $\tau = 120$ s matches KP-T as far as the coverage time is concerned, but provides a better coverage ratio. The combined maximization of contacts and coverage time can thus achieve better performance than contacts-only or time-only driven solutions even in borderline conditions.

We can draw the following conclusions:

- when the goal is to maximize contacts and coverage time, simple, hierarchical solutions that exploit knowledge of vehicular mobility can lead to quasi-optimal results over large-scale road topologies;
- the coverage time threshold τ can be used to calibrate the deployment so that it is preferably driven by vehicle-to-DP contacts or by coverage time.

5.7 Conclusions

We proposed a maximum coverage approach to the problem of information dissemination in intelligent transportation systems. The formulations and relative heuristics we presented tackle both the case in which maximizing vehicle-to-DP contacts is the only goal, as well as the case in which coverage time is also an important aspect to account for. We evaluated the different solutions in a real world topology, showing that knowledge of vehicular mobility is the main factor in achieving an optimal deployment of DPs. Our results also prove that, given such knowledge, simple heuristics can be successfully employed to plan a deployment capable of informing more than 95% of vehicles with a few DPs.

6

Understanding, Modeling and Taming Mobile Malware Epidemics in a Large-scale Vehicular Network

6.1 Introduction

Pervasive wireless machine-to-machine (M2M) communication is foreseen to be a game changer for many daily life activities, other than a technology enabling a broad range of new applications. This motivates the ever-growing availability of a wide variety of long- and short-range communication-enabled devices, from smart-phones to tablets, from notebooks to microwaves, from refrigerators to cars. The new generation of smart objects shall grant faster, cheaper communication with our friends and co-workers, easier home management, safer and more efficient mobility.

However, as it is often the case, with great profits comes high risk. If not properly secured, the network interfaces of smart devices can turn into easily exploitable back-doors, allowing illegal remote access to the information stored on the device as well as to the local network it may be connected to. Even worse, M2M communication could be leveraged by self-propagating malware to reach a large number of devices and damage them, disrupt their services or steal sensible data. The first mobile malware that spread itself through Bluetooth wireless connection, the *Cabir* worm (25), appeared in 2004

and was soon followed by several evolutions (26, 27).

The low penetration of smart devices and the heterogeneity of the operating systems have prevented major outbreaks of M2M-based worms to date (28). However, as the diffusion of communication interfaces keeps growing and the OS market becomes more stable, with two or three major competitors remaining, it is easy to predict that we may have to face smart-device worm epidemics in the future. It is thus important to understand today which are the risks we may be facing tomorrow. In fact, the behavior of potential malware in different sensible M2M communication environments has been a subject of research ever since Cabir made its first appearance. Simulative and experimental studies have outlined the risks yielded by the diffusion of so-called mobile worms via direct Bluetooth infection, both within campuses (29) and in urban areas (30), via metropolitan Wi-Fi hot-spot associations (31), and through text messaging in cellular networks (32). In all these cases, it was found that, although mobile worms propagate at speeds that are orders of magnitude lower than their Internet counterparts, they are less easily detectable and still fast enough to pose a threat.

One of the scenarios where mobile malware could cause the most damage is the automotive one. Indeed, vehicles feature today a wide range of Electronic Control Units (ECUs) interconnected by a bus, e.g., the Controller Area Network (CAN), that directly determine most of the cars' automatic behaviors. Experimental tests have proven that not only ECUs are extremely fragile to the injection of non-compliant random messages over the CAN, but that a knowledgeable adversary can exploit them to bypass the driver input and take control over key automotive functions, such as disabling brakes or stopping the car engine (33).

Lives could be thus put at stake, if a remote attack was run against a moving vehicle's ECUs. What is worse is that the above has been proved to be feasible even remotely, by exploiting the Tire Pressure Monitoring Systems (TPMS) (34) or the CD player, Bluetooth and cellular interfaces (35).

And, in that sense, the forthcoming IEEE 802.11p-based WAVE interfaces, allowing direct vehicle-to-vehicle (V2V) communication, risk to significantly widen the range of attack surfaces available to adversaries.

Despite the dangerousness of malware diffusion in vehicular environments, and through V2V communication in particular, only a few works have dealt with the topic before, mainly considering small-scale highway scenarios.

In this paper, we extend such works by proposing a more comprehensive study of the spread dynamics of a mobile worm that exploits WAVE V2V communication to self-propagate. Namely, we provide the following contributions:

- we consider a vehicular environment encompassing a geographical region of 10.000 km^2 and including more than 3.600 km of highway, regional and urban roads. Such a scenario is orders of magnitude larger than those considered in the current literature on vehicular worm diffusion, and more heterogeneous in terms of the road traffic description, comprising congested road arteries, moderately trafficked routes and underutilized rural streets at once. Deriving results in such a vast and variegated scenario allows for a better understanding of the actual level of danger of vehicular malware;

- we characterize the features of a generic vehicular worm, and assess their impact on the malware spreading and survival in the large-scale scenario above. Our analysis accounts for different WAVE technology penetration rates, time and location of the infection origin, as well as for the diverse factors that determine the worm efficiency in passing from one vehicle to another;

- we provide a model of the worm spreading speed, that builds on statistical road traffic data commonly available at transportation authorities. Such a data-driven approach (i) leads to a model that is significantly simpler than traditional mathematical descriptions of epidemics in vehicular environments, and (ii) unlike the latter, does not require unrealistic assumptions on the inter-vehicle arrival and spacing processes. More importantly, and despite its low complexity, our model proves very effective in representing the worm propagation process over the complex heterogeneous road network we consider.

- we leverage our model for the smart patching of a vehicular worm outbreak through the cellular network. Results show that our approach achieves a 100% healing of infected nodes, with unnecessary patching limited to less than 20% in the worst case.

The chapter is organized as follows: After a discussion of the related literature, in Sec. 6.2, we identify the features of a generic vehicular worm in Sec. 6.3. Simulative results on the worm spreading are presented in Sec. 6.4. Our worm spread model,

introduced and validated in Sec. 6.5, is leveraged in Sec. 6.7 for the smart containment of the worm epidemics. Finally, Sec. 6.8 concludes the paper.

6.2 Related Work

Worm epidemics. As first pointed out by Mickens and Noble (114), spreading models of human viruses from traditional epidemiology cannot be directly adapted to the diffusion of worms in mobile network environments. In fact, the worm spreading process differs even depending on the mobile network scenario considered, and, when focusing on vehicular environments, the literature is relatively thin.

The seminal work conducted by Khayam and Radha (115) introduces a first analytical study of vehicular worm spreading in a highway environment. In order to make the model tractable, their analysis relies on average values rather than on a complete description of the network connectivity. That way, however, the model fails to capture the complexity of the vehicular network topology, and overestimates the infection rapidity. This is also noted by Nekovee (116), who adopts instead a *frozen network* approach, studying individual snapshots of the road traffic. There, the worm epidemics is simulated as subsequent transfers among static vehicles within communication range. Although the vehicular density is derived through realistic microscopic mobility models, the author employs a uniform distribution of vehicles, an assumption which has been later shown not to hold in the real world (117, 118, 119).

Such a problem is overcome by Chen and Shakya (117), who also adopt frozen-network approach, but populate the snapshots according to realistic inter-vehicle spacing distributions fitted on real-world data collected by the Berkeley Highway Laboratory. The availability of such dual-loop detector data for different daily traffic conditions allows to explore the impact of daytime on the worm spreading. However, the lack of temporal correlation between the snapshots does not allow to study the propagation of worms over time; this, in turn, precludes the possibility of leveraging the model in systems where car positions change during the spreading process, e.g., in presence of roads longer than a few kilometers or of worms that take more than a few milliseconds to self-propagate. For the same reason, this technique cannot capture diffusion through carry-and-forward, where vehicles physically transport the malware until the latter can infect other cars during occasional contacts.

The diffusion-reaction and advection models employed by Hoh and Gruteser (120) describe the spatio-temporal spreading of a mobile malware. Such an approach naturally describes the system evolution over time, and thus avoids the limitations of frozen-network analyses. However, given the particular non-random nature of the road traffic mobility, calculating the diffusion coefficient in presence of realistic vehicular movements is not always possible, which limits the precision of the model and its applicability to complex heterogeneous scenarios such as the one we consider.

Our work overcomes the limitations outlined above. First, by describing the worm propagation speed rather than relying on vehicular network snapshots, the data-driven model we propose implicitly includes the time dimension. Therefore, our model can mimic the worm spreading through both multi-hop connected forwarding as well as carry-and-forwarding over temporarily network disconnections. Second, by leveraging statistical road traffic data commonly available at transportation authorities, our model does not require any complex calibration, but its single formulation is shown to closely reproduce the spreading processes for the whole space of system parameters.

Furthermore, our evaluation is conducted on significant larger scales than those considered in previous works on vehicular worm spreading. The simulation and modeling of a scenario with over 3.600 km of heterogeneous roads allows a more comprehensive analysis of the malware epidemics than those performed on a single 10-km highway corridor.

Epidemic dissemination. The spreading of generic vehicular worms can be also assimilated to the epidemic dissemination of information in vehicular networks, making the literature on the latter topic also relevant to our work. Within such a context, many analytical models have been proposed, but they all rely on the assumption that the inter-vehicle spacing is distributed deterministically (24) or exponentially (121, 122). Although the use of these distributions simplifies the mathematical modeling, real-world experimental assessments indicate that vehicle inter-distances are typically not deterministic nor exponential (117, 118, 119). In addition, some works also consider independent car speed (123, 124), breaking the well-established car-following paradigm observed in the real world (125). Finally, all of the previous works are limited to single highway scenarios.

Our simple data-driven model does not rely on the assumption that the vehicle inter-distance and speed follow mathematically tractable distributions. Also, our study

extends to a 10.000-km^2 region, much larger than those considered in the epidemic dissemination literature. Finally, as we deal with malware rather than normal content, we address worm containment, which is not a concern in epidemic dissemination.

Non-epidemic dissemination. A number of works address the problem of defining practical protocols for the efficient dissemination of information in vehicular networks, e.g., those by Lochert *et al.* (21) and Leontiadis and Mascolo (20). These are however of limited interest in the context of our work. Indeed, the goal of a malware designed for vehicular environments is to self-propagate as largely and rapidly as possible: this makes a simple and uncontrolled epidemic diffusion the obvious choice, as it guarantees minimum delays. The cost, paid in terms of overhead induced by the spreading process, is a primary concern for non-epidemic dissemination protocols, that thus add protocol complexity to reduce it. However, the overhead is very minor concern, just a (possibly even desirable) side-effect for a rapid malware whose objective is to harm the system.

6.3 Worm epidemics in vehicular networks

In this section, we characterize the features of a generic malware designed to self-propagate in vehicular environments through WAVE V2V communication.

Worms are programs that self-propagate across a communication network through security flaws common to large groups of network nodes; they are thus different from computer viruses in that the latter need the intervention of the user to propagate. Worms can be classified on the basis of several factors (126): the *target discovery*, i.e., the way they discover targets to propagate to; the *carrier*, i.e., the infection mechanism used for the self-propagation; the *activation*, i.e., the technique by which the worm's code starts its activity on the infected host; the *payload*, i.e., the set of routines undertaken by the worm, that clearly depend on the nature and objective of the attacker.

Our interest is on the worm epidemics within the vehicular network. Therefore, in this paper we focus on the first two aspects above, the type of target discovery and the kind of carrier employed by the worm, as they mainly drive the malware self-propagation process. Our study is instead activation- and payload-independent, since we do not delve into the kind of damage caused by the worm nor the motivation behind

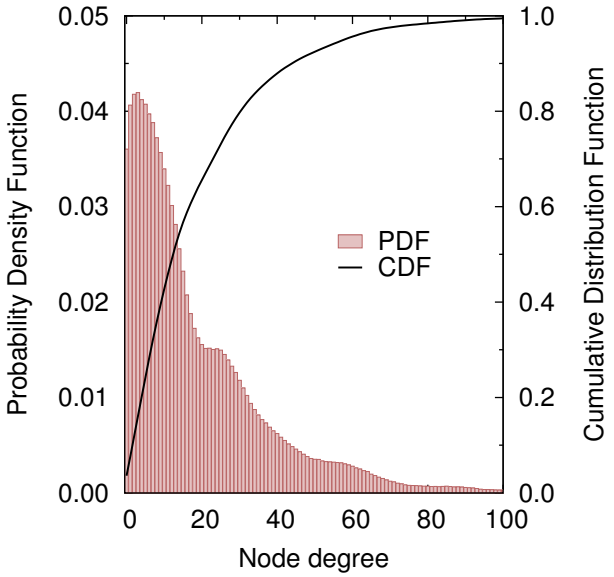


Figure 6.1: Node degree distributions in the vehicular network.

the attacks – although the discussion in Sec. 6.1 hints at how dangerous the outcome could be.

Target discovery. The target discovery in a vehicular network is dictated by the dynamics of road traffic. As a matter of fact, the relatively short range of V2V communication limits the set of potential worm's target to cars in geographic proximity of the one the worm resides in. Thus, it is the physical mobility of cars that allows the worm to enlarge its target set, by exploiting links established between communication-enabled vehicles that come into contact during their trips.

Such a *mobility-driven geographic target discovery* occurs in a way that significantly differs from that of standard Internet worms, that have to perform global or local scans for IP addresses to infect. In vehicular networks, the target discovery is implicitly (and involuntarily) supported by the forthcoming V2V communication standards, that mandate the period broadcast of beacon messages by all vehicles with sub-second periodicity: this is the case for, e.g., SAE J2735 heartbeats – part of the WAVE stack – and ETSI ITS Cooperative Awareness Messages (CAMs). A worm could then simply leverage the information collected by the vehicle via these messages to determine its current target set.

Carrier. As far as the carrier is concerned, we envision two possible carrier mech-

anisms. In the first case, the worm is designed in a way that it can self-propagate through broadcast messages, thus infecting all of its neighbors at once. We refer to this mechanism as *broadcast carrier*. In the second scenario, the worm can only propagate itself to one neighboring vehicle at a time, and we tag such a paradigm as *unicast carrier*. We argue that, in the case of a unicast carrier, no real decision has to be taken on which communication neighbor to attack: unlike what happens in the Internet, where the choice is between hundreds of millions of machines, the number of cars concurrently in range of a worm is generally low. As an example, in the scenario we will consider in our analysis, that will be detailed in the next section and whose road topology is illustrated in Fig. 6.2, the distribution of the number of one-hop neighbors (i.e., the vehicular network node degree) follows the curves in Fig. 6.1. There, we can observe how the degree ranges from a few units to a few tens of vehicles at most, and that a car typically has less than 10 neighbors half of time. Such a small target set size does not allow for an actual selection of a target node subset, and a rapid malware would simply infect all of its neighbors. This leads in the end to an epidemic spreading of the worm even in the unicast carrier case, as we will see in our analysis.

The carrier is also characterized by a second aspect, i.e., the number of transmissions (either broadcast or unicast) required to complete the infection. This value depends on the length of the worm code and on the way it is hidden in the messages. We translate this aspect to a second parameter, referred to as *carrier latency* and indicated as τ in the following. The carrier latency is the amount of time a worm needs to self-propagate to all of its neighbors (in the broadcast case) or to one neighbor (in the unicast case). We remark that τ accounts for eventual protocol-related delays, due, e.g., to association or session establishment procedures, wireless channel contention or lost message retransmissions.

SIR model. Considering the worm epidemics from the viewpoint of the whole network, and borrowing the terminology from epidemiology, in this paper we will adopt a Susceptible, Infected, Recovered (SIR) model with Immunization. According to this model, a clean node is *susceptible* to become *infected* by the worm, but it is *healed* if it receives a dedicated cure, i.e., a patch, that prevents it from contracting the infection again. The same cure can be delivered even to a susceptible node, which is then *immunized*, i.e., it cannot be infected by the worm. We also denote the first infected

vehicle as *patient zero* and its location at the time it was first infected as the *origin* of the worm infection.

The population affected by the SIR model with Immunization is formed by all the communication-enabled vehicles circulating in the geographical area of interest that suffer from the security flaw exploited by the worm to propagate. We thus characterize the population through a *penetration rate* parameter, indicated as ρ, that indicates the fraction of vehicles participating in the vehicular network and susceptible of being infected from the worm. Thus, ρ accounts for both the WAVE technology popularity and the security flaw diffusion.

6.4 Simulation results

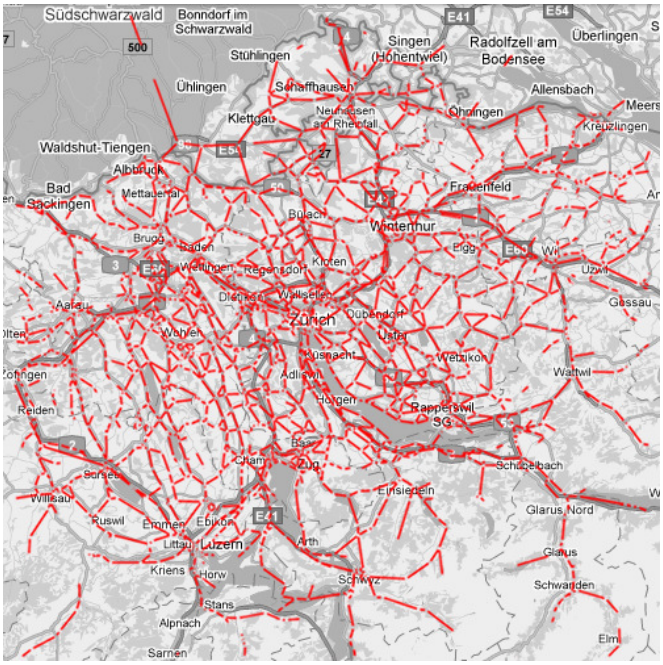


Figure 6.2: Road topology scenario, scale 1:2.700.000.

We run a comprehensive simulation campaign in order to unveil the major features of worm epidemics in large-scale vehicular networks, as well as the impact that the different system parameters have on them.

Our reference scenario encompasses the whole Canton of Zurich, an area of 10.000 km^2 in Switzerland. The region, whose 3.683-km road layout is portrayed in Fig. 6.2,

comprises the urban and suburban neighborhoods of Zurich, several smaller towns nearby, as well as the highways, freeways and minor regional roadways interconnecting them. The mobility of vehicles in the area has been synthetically generated by means of the multi-agent microscopic traffic simulator (MMTS) developed at ETH Zurich. The MMTS queuing-based mesoscopic modeling approach has been proven to reproduce real-world large-scale traffic flow dynamics and small-scale car-to-car interactions (127). That of the Canton of Zurich is in fact an unescapable choice, as it is the only mobility dataset we can leverage for a large-scale study of vehicular worm epidemics. Indeed, no other synthetic or real-world mobility dataset that is publicly available covers today a similarly wide region in a comparably realistic manner.

From a network simulation viewpoint, the scale of the scenario, where up to 36.000 vehicles travel concurrently for a time span of several hours, prevents the use of a traditional network simulator, such as ns-3 or OMNeT++. Instead, we developed a dedicated simulator, that avoids the detailed processing of messages through the whole network stack and adopts a simple R-radius disc modeling of the radio-frequency signal propagation[1]. Such a design makes simulations of very large-scale vehicular networks computationally feasible, providing significant qualitative insights into the system behavior in presence of different carrier types, penetration rates, V2V communication ranges and infection origins. All simulation results are averaged over 20 runs. Finally, we remark that for the moment our focus is on the understanding of the worm propagation in the vehicular environment. We will address malware patching later on, in Sec. 6.7. Thus, we do not introduce a cure, i.e., a patch, capable of recover or immunize the network nodes. In epidemiology, this is equivalent to consider a Susceptible, Infected (SI) model. We will address the patching SIR process in Sec. 6.7.

6.4.1 Worm carrier

We first study the impact of the worm carrier. Let us first assume that patient zero originates in downtown Zurich, i.e., at the center of the map in Fig. 6.2 approximately, at 3 pm, when the road traffic intensity is at its peak. In Fig. 6.3, we focus on the carrier mechanism, setting the carrier latency τ at 1 s and comparing the results achieved by a broadcast carrier against those obtained by a unicast carrier. The performance is evaluated in terms of *infection ratio*, i.e., the fraction of vehicles the worm has infected

[1]Our simulator is available at `http://trullols.site.ac.upc.edu`.

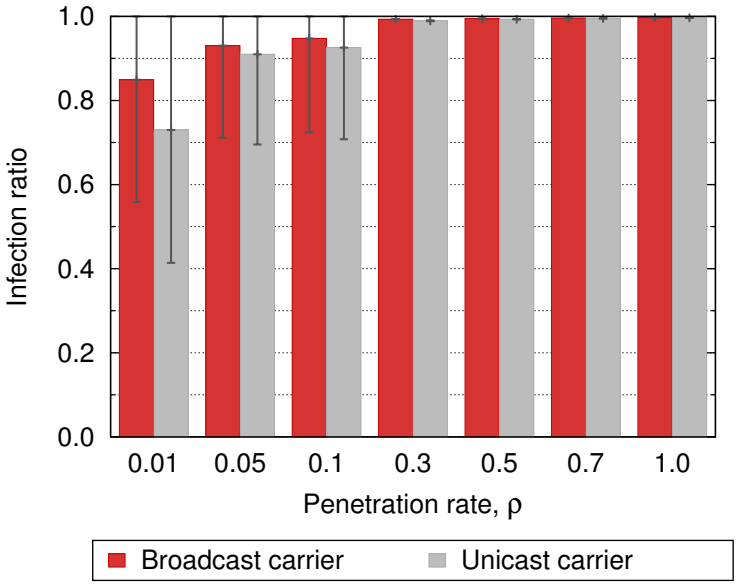


Figure 6.3: Carrier mechanism: Broadcast versus unicast carrier, in terms of infection ratio and under different WAVE penetration rates, ρ. Error bars represent the standard deviation.

after four hours from the injection time[1]. The x axis reports the combined WAVE technology and security flaw penetration rate ρ, that grows from 0.01 to 1. i.e., from 1% to 100% of the vehicles. Error bars represent the standard deviation.

We observe that a broadcast carrier achieves a higher infection ratio than a unicast one. This is expected, since the latter mechanism requires the worm to self-propagate multiple times, each requiring a time τ, to reach all the nodes that a broadcast-carrier worm can reach with a single infection in a time τ. However, the difference is noticeable at very low penetration rates only, since the two carrier mechanisms perform basically the same once 5% or more of the automobiles are susceptible of contracting the worm. Furthermore, even for $\rho \leq 0.05$ the performance gap is marginal.

As far as the impact of ρ is concerned, higher penetration rates clearly lead to a more connected network of susceptible vehicles, which in turn facilitates the spreading of the worm. However, it is surprising to note how very high infection ratios are achieved even in very sparse networks comprising 1 to 5 percent of the cars. In fact, a $\rho = 0.3$ is largely sufficient to achieve a complete infection of the network. This

[1] As we will see, four hours are largely sufficient to our analysis.

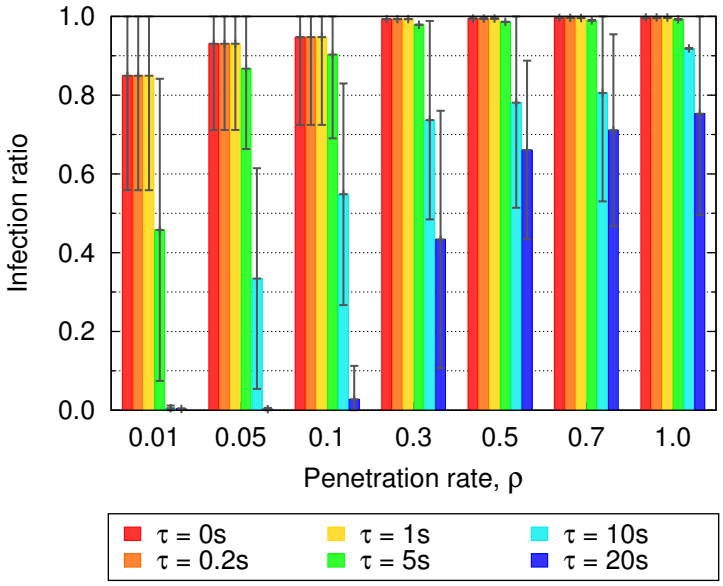


Figure 6.4: Carrier latency,τ: Impact of diverse carrier latencies versus ρ.

phenomenon is imputable to the fact that the high velocity of cars can compensate for the reduced penetration rate, generating many V2V contacts and facilitating the worm self-propagation in a carry-and-forward fashion.

In Fig. 6.4, we focus on the broadcast carrier case and study the impact of the carrier latency τ, in presence of different penetration rates. More precisely, we consider a τ ranging from 0 s (which represents an ideal upper bound to the worm spreading performance, since the worm infection is instantaneous) to 20 s, and assume ρ to vary between 1% to 100%.: an infected node can contaminate with a single broadcast transmission its whole 1-hop neighborhood. The plot shows, for each combination of penetration rates and τ, the ratio of vehicles in the 100×100 km^2 region that have been infected by the worm. In addition, as reported in the legend, different shades identify the time required for the worm to infect a given portion of the vehicular network: full-saturation colors thus correspond to a quick worm spreading, taking just a few minutes, while, as bars tend to white, the spreading time becomes longer. Spreading times of half an hour or more are in white. We remark that, even in presence of low penetration rates, a sufficiently fast worm (i.e., one capable of infecting its 1-hop neighborhood in one second or less) can still successfully infect the vast majority of the vehicles in a very large region such as the one we considered. In fact, even when $\tau \leq 1$ s, at least 85% of

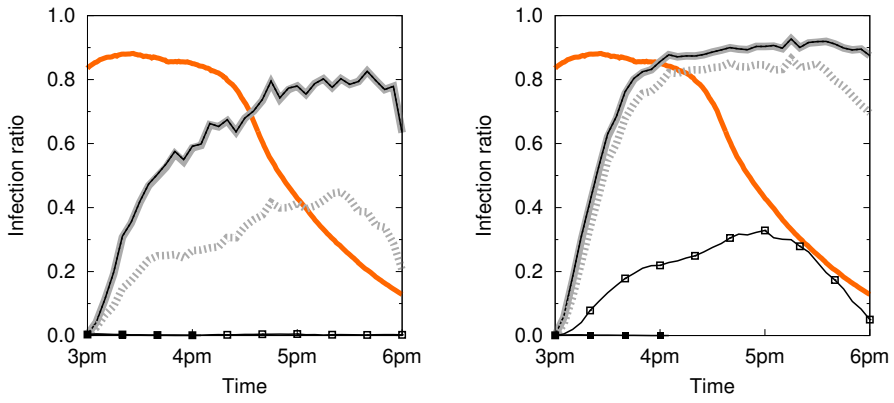
the vehicles are infected under any penetration rate. Longer carrier latencies appear instead to be more dependent on ρ: when $\tau \geq 10\,\mathrm{s}$ the worm is unable of infecting the whole network even when all the vehicles are susceptible of contracting it.

The observations above let us conclude that: (i) unicast-carrier worm are as dangerous as broadcast-carrier ones; (ii) worms do not need a lot of vehicles to successfully spread through a large area, due to the fast dynamics of road traffic that tend to facilitate the malware propagation; (iii) worms do not need to be extremely fast in infecting neighboring vehicles, as a 1-second carrier delay (a perfectly realistic value, considering that worms occupy a few tens of KBytes of code and the V2V basic data transfer rate is in the order of a few Mbps) proves to be largely sufficient to vehiculate the worm to the whole network in all conditions.

6.4.2 Worm epidemics over time and survivability

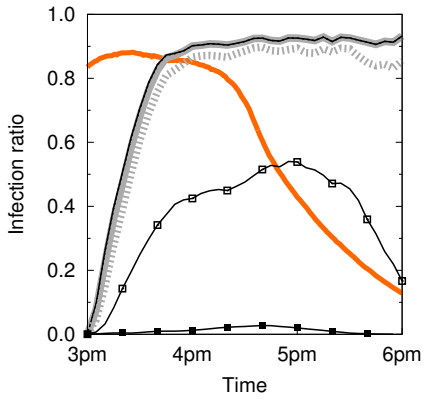
The percentage of infected nodes is not the only metric of interest in the analysis of the worm propagation. The time needed for the worm to reach different regions of the vehicular network is also an important factor. Another relevant aspect is the *worm survivability*, defined as the period of time during which the infection can self-sustain in the vehicular network. These metrics can be studied by observing the dynamics of the epidemics over time.

In Fig. 6.5, each plot refers to a specific penetration rate ρ, and portrays the evolution of the infection for different values of the carrier latency τ. When $\rho = 0.01$, in Fig. 6.5(a), only rapid malware with carrier latencies τ of 1 s or less can propagate through most (although not all) of the network. The bell-shaped infection ratio for $\tau = 5\,\mathrm{s}$ is explained by the aggregated road traffic volume, also depicted in the figure: the worm is not fast enough to infect the whole network before the traffic peak ends, at around 4.30pm, i.e., 1 hour 30 minutes after the infection started. As a result, the infection stays limited to the surroundings of the injection area, and then dies out when the traffic becomes sparser due to vehicles leaving the area or stopping. Slower worms do not even start to spread in the system. Increasing ρ to 0.05, in in Fig. 6.5(b), also allows slightly slower worms, characterized by a τ in the order of a few seconds, to infect an even larger majority of the vehicles. Namely, worms with $\tau \leq 5\,\mathrm{s}$ perform similarly and achieve a 95% infection ratio with a linear growth during the first 45 minutes from the worm injection. This clearly makes such worms extremely dangerous,

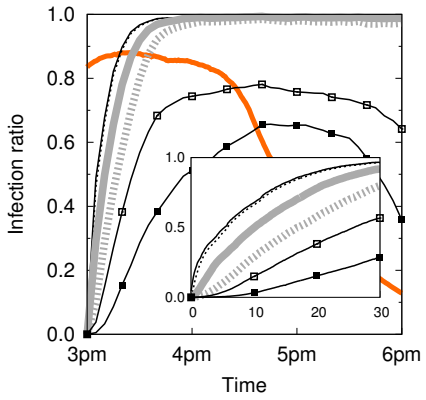


(a) $\rho = 0.01$

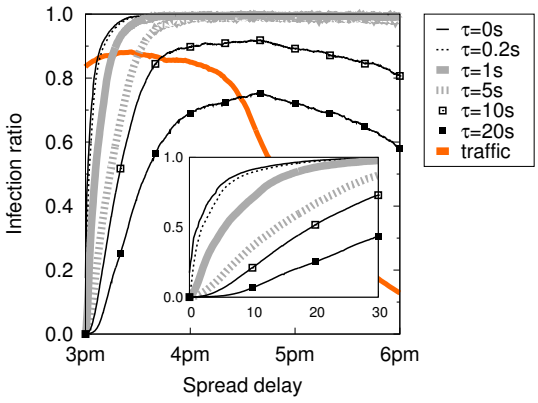
(b) $\rho = 0.05$



(c) $\rho = 0.1$



(d) $\rho = 0.5$



(e) $\rho = 1.0$

Figure 6.5: Worm epidemics and survivability as a function of the penetration rate, ρ, using the broadcast carrier mechanism.

since, in order to be effective, a patch should be provided to network nodes within the very few minutes after the worm injection. The bell shape now characterizes the diffusion of malware with a carrier latency of 10 s, for the same reasons discussed above. Slower worms find it still difficult to spread at such a low ρ.

A larger participation of 10% of the cars in the network, in Fig. 6.5(c), does not affect the behavior of worms characterized by a $\tau \leq 5$ s, and only favors slower worms. On the other hand, as the population of susceptible vehicles grows to 50% of the overall road traffic, in Fig. 6.5(d), we remark two effects. First, the infection evolutions of the faster worms start to separate, as highlighted in the inset plot, which details the spreading process during the first 30 minutes from the worm injection time. Indeed, very fast worms (i.e., with $\tau < 1$ s) were previously limited by the lack of multi-hop connectivity, and had to rely on carry-and-forward to find new susceptible vehicles. As a result, their performance, hitting the bar imposed by the limited network connectivity, was similar to that of slower worms (e.g., with $\tau = 5$ s). Now, fast malware can take advantage of the presence of larger connected clusters of vehicles, and spread over 95% of the network in some 20 minutes. Moreover, the growth is now faster than linear, with 50% of the nodes being infected in less than 6 minutes. As a second remark, the higher penetration rate has a largely beneficial effect on the spread dynamics of slower worms, as now the curves for $\tau \geq 10$ s depict the infection of very wide portions of the network. However, we can still notice that the worm does not self-sustain, since its infection rate tends to drop once the traffic peak ends.

In Fig. 6.5(e) we consider the case where all the vehicles are communication-enabled, i.e., a maximally connected network. The effects already observed in the previous plot are here exacerbated, with the faster worms capable of reaching 50% of the network in less than 2 minutes and spreading over 95% of the network in 10 minutes. Slower worms also take advantage from the increased network connectivity, although not yet reaching a complete infection ($\tau = 5$ s) or even self-sustainability ($\tau \geq 10$ s).

A similar temporal analysis can be done when comparing different carrier mechanisms. Fig. 6.6 depicts the infection evolution of broadcast and unicast worms, in presence of varying penetration rates, when $\tau = 1$ s. The inset plot shows again the detail of the first thirty minutes of the spreading process. We can note that, unlike what seen for the final infection ratio in Fig. 6.3, unicast and broadcast carriers differ in terms of delay. However, such a difference is mostly remarkable at high penetration

rates. As the penetration rate decreases, the difference in the time needed to infect the network is reduced, and the two paradigms match when 10% or less of the cars are involved in the network.

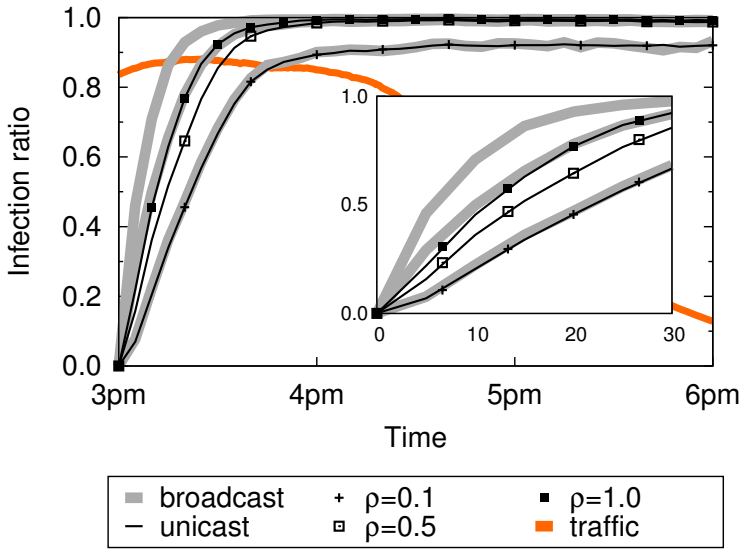


Figure 6.6: Carrier mechanism: Worm epidemics and survivability versus the carrier mechanism.

6.4.3 Carrier latency and contact time

The previous results show a striking difference in the spread process of worms characterized by various carrier latencies. In particular, values of τ of 1 s or less seem to result in high infection rates no matter the number of vehicles involved in the network; moreover, such values of τ allow the infection to occur much faster as the penetration rate increases. On the other hand, worms with a $\tau \geq 10$ s need high penetration rates to diffuse and take a lot of time to do so. Values of τ in between those seem to result in intermediate behaviors.

The physical reason behind these performance lies in the vehicle-to-vehicle contact duration distribution, in Fig. 6.7. Most contacts among moving vehicles are very short: more than 70% of them last less than 5 seconds, and less than 10% of the contacts are longer than 10 seconds. However, the distribution is heavy-tailed, with a 5% of the contacts lasting one minute or more, as from the inset plot. Our conclusion is that fast

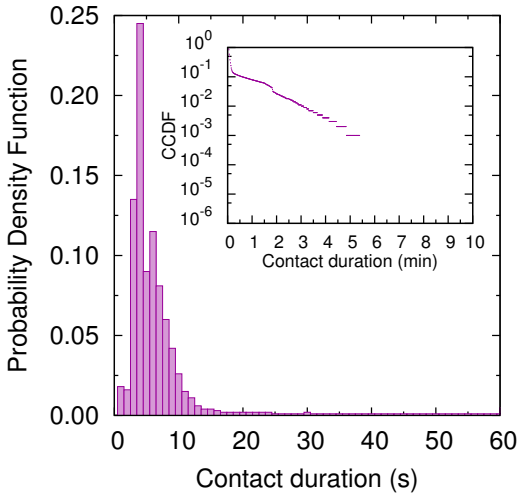


Figure 6.7: V2V contact duration distribution.

worms, capable to spread from one vehicle to another in one second or less, can exploit any contact occurring in the network. Conversely, a worm characterized by a τ of 5 s will be only able to leverage 20% of the contacts, and one with $\tau = 10$ s will propagate through a mere 8% of the actual V2V links. In other words, fast worms enjoy a more connected network to spread through.

6.4.4 Summary

Summarizing our findings, we can conclude that, no matter the penetration rate and carrier mechanism considered, a reasonably fast worm can be extremely dangerous. More precisely, we observed how a worm that fits a few IP packets, and that could thus be transmitted over the wireless medium in less than one second (accounting for channel contention and losses), can easily infect a vast majority of the tens of thousands of vehicles traveling in a very large urban, suburban and rural region. Even worse, such infection would occur in a time in the order of few tens of minutes at most, making it hard to counter the infection. The physical reason behind such an impressive performance of the worm diffusion lies in (i) the high number of short-lived connections generated by the movement of vehicles, and (ii) the elevate mobility of nodes in the vehicular network. Both factors contribute to create an ideal environment for a fast worm to self-propagate. We also conducted tests on the impact of the infection origin,

in terms of geographical location and injection time. Although we omit[1] these results due to space limitations, we found that the location of patient zero has a dramatic effect on the epidemics, due to the spatial heterogeneity of road traffic. Conversely, the injection time only has a minor impact on the worm survivability.

6.5 Modeling the worm epidemics

The simulation results presented in the previous section illustrate the epidemic behavior of a generic vehicular worm in a large-scale scenario. In this section, we propose a model capable of faithfully mimicking such a behavior. Our broadcast-carrier worm propagation model is data-driven, in that it is based on commonly available road traffic statistics. Although simple in its expression, the model can capture the exact impact of the wireless communication radius R, the penetration rate ρ and the carrier latency τ on the large-scale worm propagation delay. In addition, and unlike many of the models discussed in Sec. 6.2, our model does not require any assumption on the distribution of vehicle inter-spacing, speed or inter-arrival time (IAT). For the sake of completeness, we mention that we found the latter to be an exponential/normal mixture in the Canton of Zurich dataset, as portrayed in Fig. 6.8. This result matches the real-world observations in (118), and invalidates the typical deterministic and Poisson arrival assumptions employed in the literature.

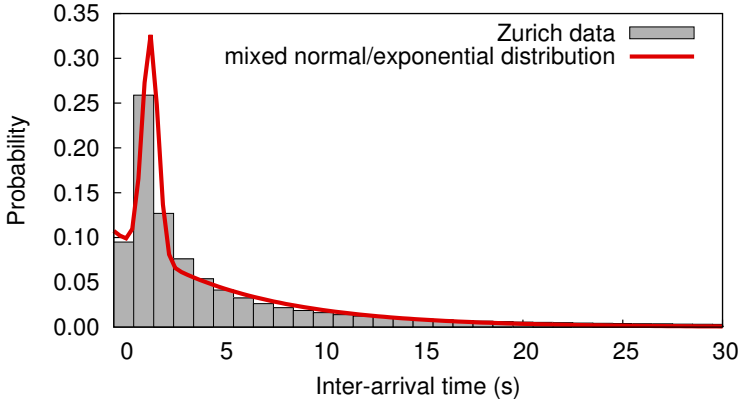


Figure 6.8: IAT distribution in the ETH Canton of Zurich dataset.

The reason is that recent experimental evaluations have shown that the exponential

[1]For a detailed description, see Technical Report UPC-DAC-RR-2012-19 at https://www.ac.upc.edu/app/research-reports/html/2012/19/report.pdf

IAT assumption does not hold in real-world highway settings (118). The IAT has instead been proved to follow a mixed normal-exponential distribution of the form

$$f_{IAT}(t) = w_N \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(t-\mu)^2}{2\sigma^2}} + w_E \lambda e^{-\lambda(t-\nu)}, \tag{6.1}$$

where μ, σ, λ and ν are the parameters of the normal and exponential distributions, while w_N and w_E are the weights associated to each distribution. Interestingly, we found this result to apply to the Zurich scenario as well. When computing the per-road IAT distribution in our mobility dataset, we do not obtain an exponential curve, rather the mixed distribution suggested in (118), shown in Fig. 6.8 with the associated fitting curve, for which μ=1.326 s, σ=0.391 s, λ=0.165 s^{-1}, ν=1.424 s, w_N=0.2426 and w_E=0.7574.

As a consequence, we cannot rely on previously proposed analytical models for worm or data propagation along roads. Instead, we provide a novel formulation, that accounts for the realistic nature the vehicular mobility considered in our study. The model we propose builds on (i) information about the road topology and (ii) statistical information about the road traffic. In the first category, we need, for each lane i, knowledge of its length l_i and a list of the other roads it intersects with: data that can be easily extracted from road map services such as, e.g., OpenStreetMap. As for the second category, the model requires information on the average travel speed $v_i(t)$ on a road lane i at time t and mean inter-arrival time $a_i(t)$ at road lane i and time t. In our case, we extract such information from the vehicular mobility dataset. In real-world applications, Such road traffic metrics are commonly collected by transportation authorities and automobile service operators through induction loops, infra-red counters, traffic monitoring cameras, and, more recently, floating car data systems (128). Therefore, such historical or statistical data is currently available for large portions of the road topology and its public disclosure is growing, fostered by open data initiatives.

Road traffic information is by its own nature time-varying, i.e., the average speed and IAT are not the same during the day or on different days of the week, which is why we consider $v_i(t)$ and $a_i(t)$ be dependent on time. The aforementioned statistical data necessarily reflects this aspect, with a finite yet representative time granularity[1]. Also,

[1]In our evaluation, we assume that statistical data on the road traffic is aggregated and updated with a time granularity of 15 minutes, largely sufficient to capture the time variability of the Zurich road traffic metrics.

note that although higher order statistics may be available, our model only requires knowledge of the mean values of $v_i(t)$ and $a_i(t)$ at each time period. Leveraging the data above, we next discuss the modeling of the worm propagation speed along a single road, in Sec. 6.5.1, and then extend the result to a network-wide worm propagation, in Sec. 6.5.2.

6.5.1 Per-road worm propagation

Our goal is initially to model the *worm propagation speed*, $s_i(t)$, along a lane i characterized by average road traffic parameters $v_i(t)$, $a_i(t)$ at time t, accounting for the technology-related parameters R, ρ and τ. For the sake of clarity, in the following we refer to a generic time instant and drop the time notation. We start from the consideration that the worm propagation speed mainly depends on the network connectivity level. Namely, the malware can propagate wirelessly, and thus at a high speed, in a well-connected vehicular network where multi-hop communication can take place. Conversely, the worm propagation is slowed down when communication opportunities are scarce. Focusing on the two extreme cases, we can state that: (i) in complete absence of vehicle-to-vehicle connectivity, the worm propagates at the vehicular speed v_i, as it is physically carried by isolated cars; (ii) in presence of a complete road coverage by a very dense multi-hop vehicular network, the worm instantaneously[1] jumps of a distance equal to the communication range R at each carrier latency, the latter requiring a time τ during which the worm still moves at the vehicular speed v_i. Therefore, the worm propagation speed has an expression of the type distribution of the form

$$s_i = v_i + \frac{R}{\tau} f(a_i, v_i, \rho, R, \tau) \tag{6.2}$$

where $f(\cdot)$ is a function of the different system parameters that represents the vehicular network connectivity level. Such a function assumes values between 0 (absence of connectivity) and 1 (fully connected network). In order to characterize the exact expression of $f(\cdot)$, we observe the impact of the different parameters on the network connectivity. Considering the simplified case of vehicles moving along one single road direction, the average distance between two subsequent vehicles is given by $a_i \cdot v_i$,

[1] We assume the RF signal propagation delay to be negligible, since it is in the order of nanoseconds, a value at least three orders of magnitude lower than the duration of the other events involved in the process.

i.e., the distance traveled by the first vehicle before the following one enters the same road. The technology penetration rate can be accounted for by assuming that the first vehicle is equipped with a communication interface, and ρ can be seen as the probability that the following vehicle is communication-enabled as well. Then, an average of $1/\rho$ vehicles must enter the road before a second car equipped with a radio interface actually appears on the road. The average distance between to vehicles that are participating in the network is then $\frac{a_i v_i}{\rho}$. The connectivity is determined by the relationship between the distance above and the transmission range R. In particular, it is the ratio between the two values, $\frac{a_i v_i}{\rho R}$, that matters: the lower the ratio, the higher the network connectivity, and vice-versa.

The discussion above refers however to the case of vehicles all moving in a same direction. The presence of an opposite vehicular flow can be accounted for through a factor K, that divides R. In other words, if vehicles in the other direction of movement can be leveraged for the worm propagation, a range R that is K times smaller provides the same connectivity achieved by a range R in a single-direction scenario. Alternatively, the introduction of K can be interpreted as the fact that one can allow a distance K times larger between two communication-enabled vehicles and still achieve the same level of connectivity. As discussed later, we found the value of K to be almost invariant to the whole range of road traffic and communication settings we evaluated, and we thus treat it as a constant in the following. Finally, τ has no major impact on the network connectivity expressed by $f(\cdot)$, since it is an application-level parameter. The only case where τ can indirectly affect the network connectivity is that of a carrier latency so large to be comparable to the time required to travel along a whole road segment between two intersections. However, the latter is at least several tens of seconds, while the values of τ of interest to our study are significantly shorter. Therefore, we neglect the impact of τ in the following.

Summarizing our discussion above, the worm propagation speed can be expressed as

$$s_i = v_i + \frac{R}{\tau}\, f\left(\frac{a_i\, v_i}{\rho\, R/K}\right). \tag{6.3}$$

We still have to identify a proper expression for the function $f(\cdot)$ and a value for the parameter K. To that end, we employ the data from our worm propagation simulations. Interestingly, by fitting the expression in (6.3) to the data, we observe that a single

function $f(x) = exp(-x^2)$ and a single value $K = 3$ fit the data for any combination of the road traffic and communication parameters. Therefore, the worm propagation speed along lane i can be finally expressed as:

$$s_i = v_i + \frac{R}{\tau}\, exp\left[-\left(\frac{a_i\, v_i}{\rho\, R/3}\right)^2\right]. \tag{6.4}$$

This single simple expression can thus be employed to describe the average worm propagation speed along any road in the Zurich scenario, given that its road traffic statistics, i.e., the average vehicular speed v_i and the average inter-arrival time a_i, are known. The equation in (6.4) allows then to evaluate the impact of the propagation settings R, ρ and τ, since it holds for any combination of the same.

Examples of the correctness of the model are provided in Fig. 6.9. Plots in Fig 6.9(a), (b) and (c) aggregate the results for roads with similar average vehicular speeds, ranging between 11 m/s (less than 40 km/h) and 28 m/s (over 100 km/h). Each plot displays a scatterplot of the worm propagation speed measured at each lane i, versus the road average inter-arrival time, a_i, with baseline parameters $R = 100\,\text{m}$, $\tau = 1\,\text{s}$ and $\rho = 1$. The red curve represents the average behavior observed over all roads, while the black curve is the result provided by our model. The plots in Fig 6.9(d), (e) and (f) show instead the worm propagation speed for different values of R, ρ and τ. There, for the sake of clarity, the scattered simulation samples are not drawn and only the average curves are reported. It is however clear that our data-driven model can faithfully mimic the average behavior of the worm propagation speed, in any road traffic condition. Of course, the model does not capture the random variability around the mean that is observed for specific roads. This is due to the fact that we only consider the average values of v_i and a_i in our study, and not higher-order moments of their distributions. This is, however, an intentional choice, that allows us to keep the model very simple, still obtaining excellent results when considering the network-wide worm propagation process, as discussed in the next section.

6.5.2 Road network-wide worm propagation

We now leverage the worm propagation speed expression in (6.4) to describe the propagation process over the whole road network. In particular, the propagation is characterized in terms of *spread delay*, i.e., the time that a worm takes to reach a specific location after the infection of patient zero.

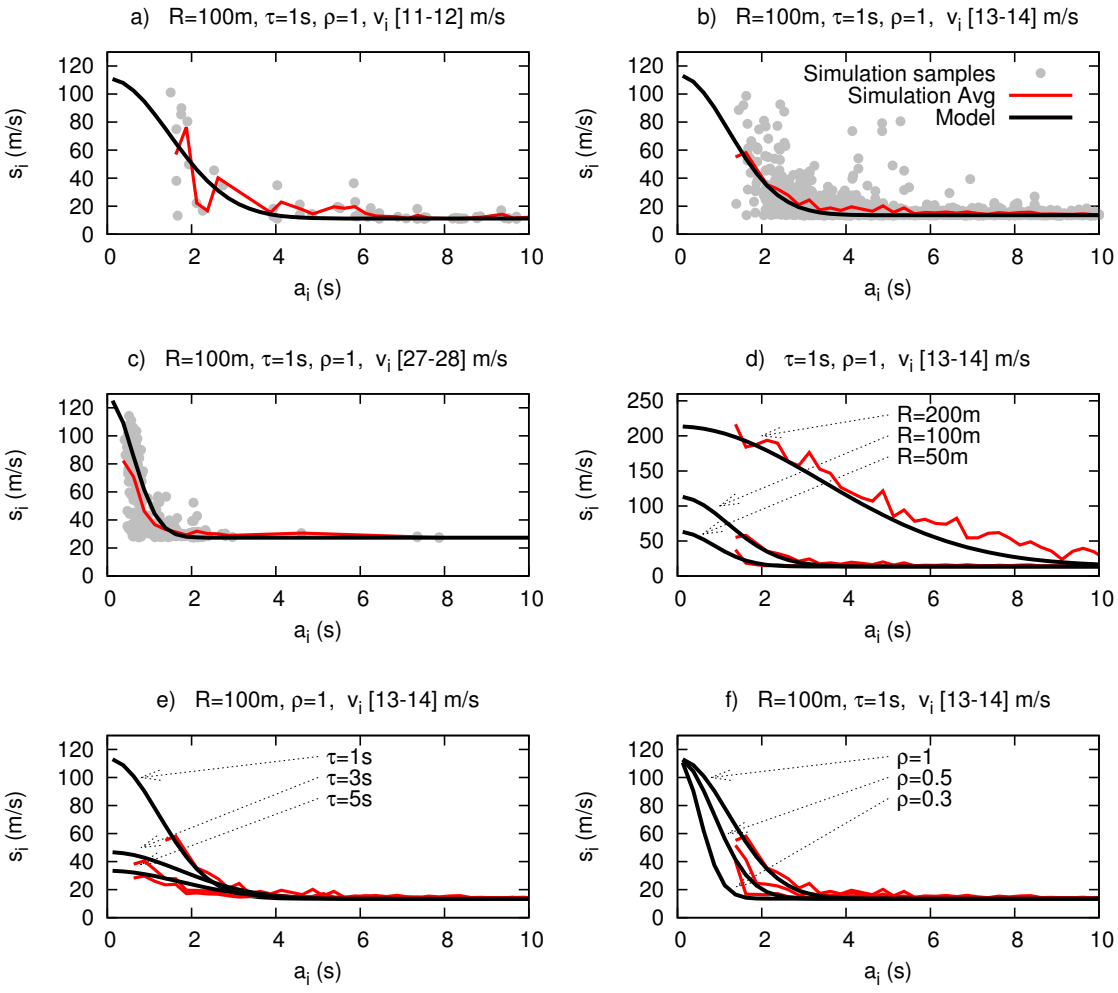


Figure 6.9: Per-road worm propagation speed, s_i, for different combinations of the road traffic parameters v_i, a_i and technology parameters R, ρ, τ.

Let us first represent the road layout as a graph $G{=}(\mathcal{V}, \mathcal{E})$, where the set of vertices $\mathcal{V}$ represents the intersections and the set of edges $\mathcal{E}$ represents the roads joining such intersections. Knowing the worm propagation speed s_i along a road segment i, the spread delay from one end of the road to the other can be derived as $w_i{=}\frac{l_i}{s_i}$. Each edge in $\mathcal{E}$ is then associated to a weight matching its spread delay w_i. Note that the resulting weighted graph is time-varying, since the worm propagation speeds along each road, and thus the weights derived from them, change over time.

Given the infection time t and the location on the road topology of patient zero, calculating the spread delay from the origin point to any other point of the region

reduces to a single-source shortest path problem on the weighted graph associated to time t. A standard Dijkstra's algorithm can be used to rapidly solve the problem. Then, the spread delay to a given location on the road network is given by the cost of the shortest path to its corresponding vertex or edge on the graph. Indeed, such a cost maps to the sum of the spread delays along the fastest path from the infection origin up to the location of interest.

Note that, if the injection point or the point at which the delay is measured are located along a road, splitting the corresponding edge and inserting a vertex is sufficient to solve the problem.

An intuitive example of the model accuracy is provided in Fig. 6.10. The plots represent the geographical propagation of the worm, as occurring in simulation and as predicted by the model. We can note that the worm propagation is almost identical in the two cases, as most of the reached points are covered at the same time by the simulation and the model. The differences, evidenced as the points that are reached at each time instant by the simulation or by the model only, are minimal and always limited to the rim of the propagation process.

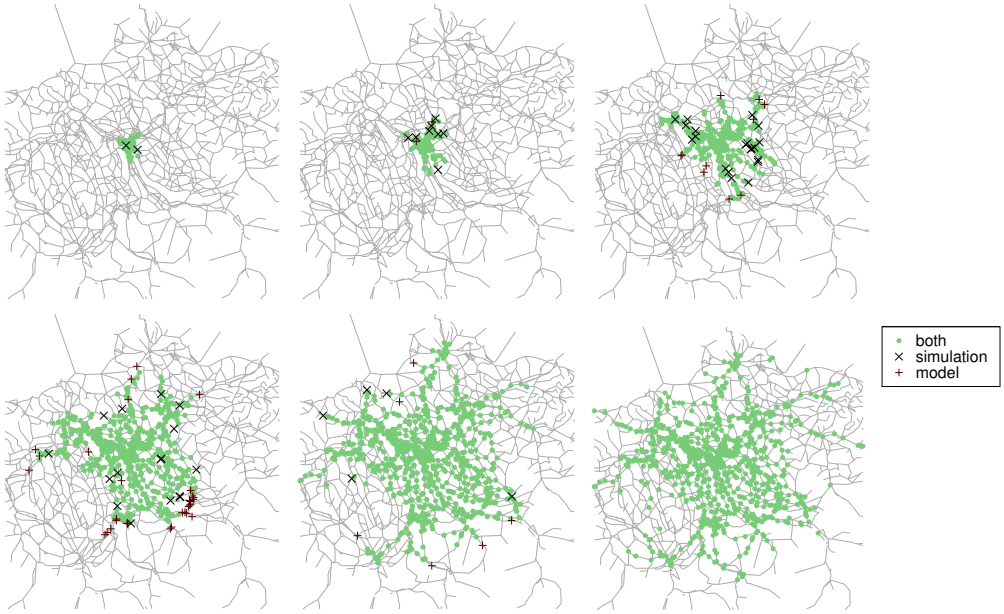
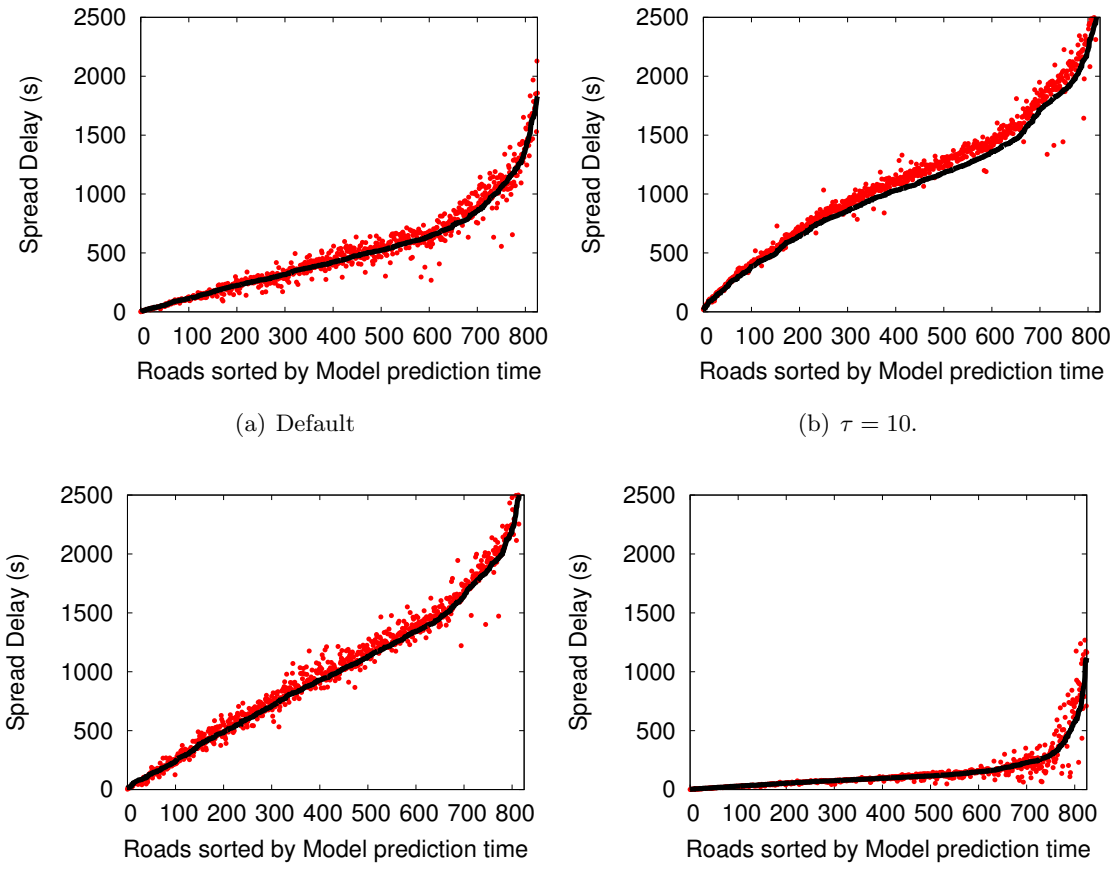


Figure 6.10: Worm propagation, as occurring in simulation and as predicted in our model with $R = 100$ m, $\tau = 1$ s, $\rho = 1$. Snapshots refer to 1, 2, 5, 10, 20 and 40 minutes after the worm injection.



(a) Default

(b) $\tau = 10$.

(c) $\rho = 0.3$.

(d) $R = 300$ m.

Figure 6.11: Road network-wide spread delay with (a) $R = 100\,\text{m}$, $\tau = 1\,\text{s}$, $\rho = 1$, and when (b) $\tau = 10\,\text{s}$, (c) $\rho = 0.3$, (d) $R = 300\,\text{m}$.

The qualitative evaluation of the model is presented in Fig. 6.11, where each plot portrays the road network-wide spread delay measured in simulation (dots) and computed by our data-driven model (solid line). Graph vertices (i.e., road intersections) are ranked along the x axis according to the worm spread delay determined by the model. Fig.6.11(a) refers to the case of $\tau = 1\,\text{s}$, $\rho = 1$ and $R = 100\,\text{m}$. Although there are a few outliers, a vast majority of the delays needed to reach the different road intersections in simulation is correctly reproduced by the model. We can observe that the quality of the result is the same when the different systems parameters τ, ρ and R are varied, in Fig. 6.11(b), Fig. 6.11(c) and Fig. 6.11(d), respectively.

A more complete picture of the model reliability is provided in Fig. 6.12, that shows

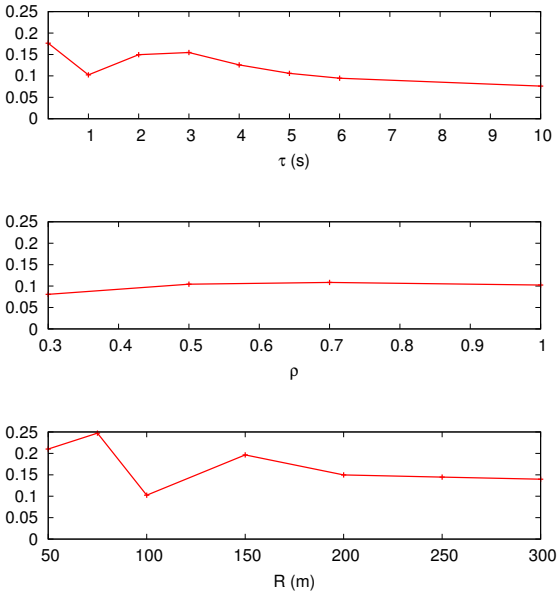


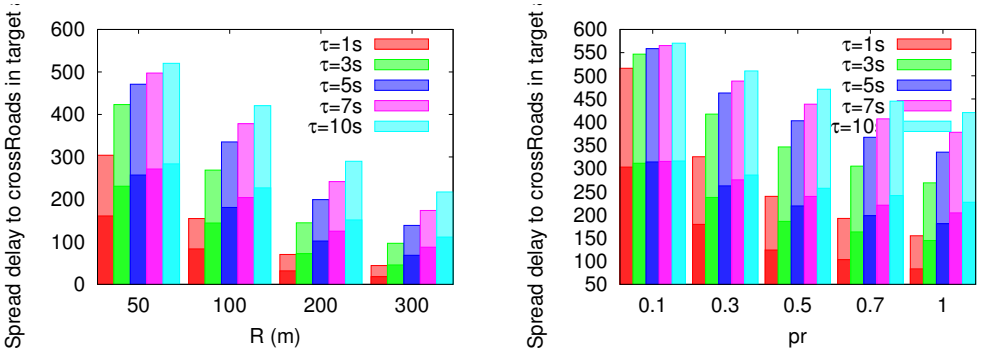
Figure 6.12: Relative Error, η, versus the carrier latency τ (left), penetration rate ρ (middle) and communication range R (right).

the average relative error η between the simulation results and the model outcome for the whole parameter space. Notably, the error remains below 0.18 for all values of $\tau \in [0.2, 10]$, stays below 0.1 for any value of the penetration rate ρ, and is at most 0.25 for short communication ranges below 50 m. The reason is that with short radio ranges, the variability of the worm propagation speed is higher for short interarrival times.

As a summary, we can conclude that in general, the fitting model approximates quite well the simulated data, even with the high variability of the data considered. Moreover, the average relative error is kept between the interval $\eta \in [0.1, 0.15]$ for most of the cases.

6.6 Model exploitation

The fitting model can be exploited to obtain several measures that are hard to obtain from the complexity point of view via simulation. As stated, knowing a road topology and simple derived transportation parameters, the per-road worm propagation speed can be easily derived. So, while acquiring all the mobility data can be very expensive, and for each parameter we want to check, running the simulations can take hours, using



(a) $\rho = 1$.

(b) $R = 100$.

Figure 6.13: Spread delay.

the model with Dijkstra algorithm, we obtain an accurate result in seconds.

6.6.1 Impact of the worm propagation parameters.

Let us assume a road topology in which the a_i and v_i are known for each $i = 1, 2, ..., E$ and the worm injection place is fixed. Fig. 6.13 shows the spread delay as a function of the ρ and R for several infection transmission times when the focus of the patient zero is located in the center of the Zurich downtown. The worm is propagated to a 10km×10km area and a 20km×20km area (transparent bars). Fig. 6.13(a) shows the spread delay for several τ and R. The figures are consistent with those ones obtained in section 6.4; low values of τ and high values of R produce fast worm propagations, while, large values of τ and low values of R slow down the propagation of the worm. Low penetration ratios produce high spread delays as predicted by eq.(6.4). Low penetration ratios imply that the average spread speed r_i tends to the average vehicle speed v_i.

6.6.2 Impact of the location of the patient zero.

The fitting model allows a more exhaustive study of the impact of the patient zero. Let us assume that an attacker wants to infer where is a good place to initiate a worm attack. Fig 6.14(a) shows the spread delay taking as origin any one of the vertices of the graph. They are ranked in increasing order of distance to the downtown center. As it can be observed, there are some roads that are far away from the center, e.g. ranked at positions 400 to 600 but that produce similar spread delays as roads that are near

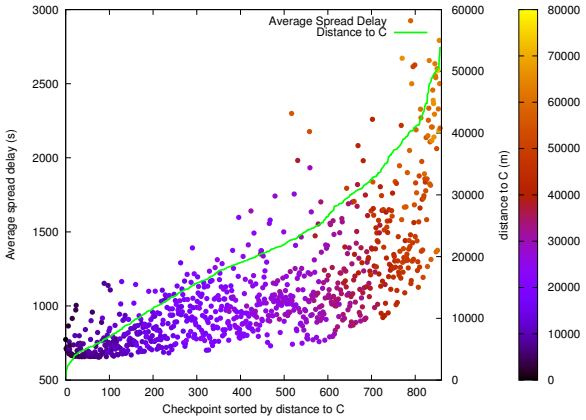
the center. On the other hand, it is clear that roads in remote areas and then distant from the downtown in general produce high spread delays. Then, we can conclude that in this road topology being near the downtown center in general produces good spread delays, but this is not so conclusive since many roads not near the center also produce good spread delays.

An attacker can figure out that those roads with good vehicle densities should be good candidates for placing the worm. Fig 6.14(b) shows the spread delay sorted by increasing vehicle density. In general, having a high vehicle density produce low spread delays. However, there again is a high number of roads with low vehicle densities that produce low spread delays. The reason is that there are roads with low vehicle densities possibly connected to high density roads, that will spread the worm fast.

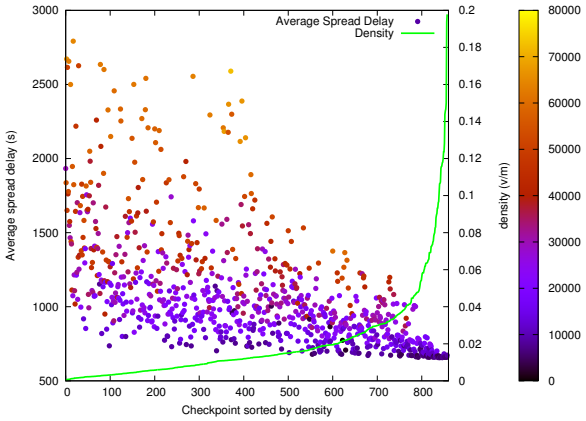
Then, a road even with low vehicle density connected to a high vehicle density area can produce fast worm propagation. Fig 6.15 shows the spread delay of a worm propagated at any road ranked taking into account the vehicle density in N-hops far away from the focus of the infection. Fig 6.15(a) groups 2-hop roads and Fig 6.15(b) groups 5-hop roads. We can observe that a road that in the 2-hop or 5-hop plots produces long spread delays is a road in a low vehicle density area and thus is not eligible to produce a fast worm propagation. On the other hand, the 5-hop plot shows us how a low vehicle density road belonging to a high vehicle density area can be elected as a good focus even having a low 1-hop vehicle density. The conclusion is that electing a patient zero in a road with neighboring roads with high vehicle densities improves the worm propagation since produces a fast initial expansion that speeds up the worm propagation.

6.7 Containing the worm epidemics

The results in the previous sections outline the dangerousness of vehicular worms, and motivate the development of solutions for the rapid containment of their outbreaks. The typical techniques proposed in the literature are preemptive immunization and interactive patching (115, 116). In the first case, a subset of the vehicles is preemptively immunized so as to prevent the propagation of the worm. However, preemptive immunization makes sense only in the case of frozen networks, where immunized nodes can disrupt the network connectivity exploitable by the malware. In presence of a more



(a) distance to the center.

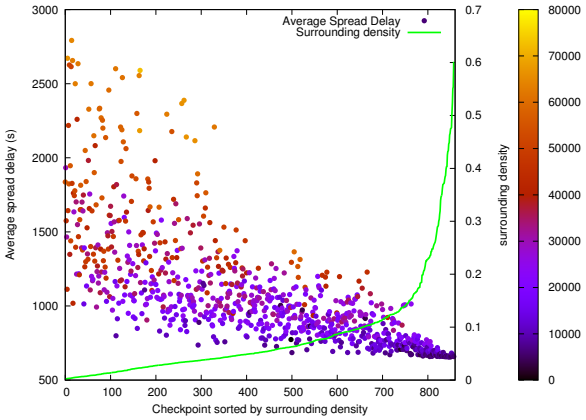


(b) vehicle density.

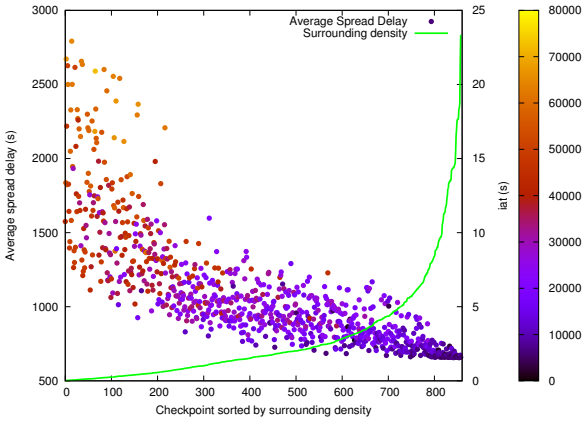
Figure 6.14: Spread delay ranked by distance to the center and by vehicle density.

complete time-evolving analysis, worms can easily overcome the obstacle of preemptively immunized vehicles thanks to the car mobility over time. In the case of interactive patching, a patch to the worm is released in the network and diffused through V2V communication in an epidemic fashion. In other words, the patching follows a spreading similar to that of the worm itself. However, resorting to V2V communication to contain the malware epidemics does not seem a sensible choice, as it implies high delays and a probability of success that cannot be certain.

We consider instead that cellular communication can be leveraged to distribute the patch in a rapid and reliable manner. Indeed, vehicles already start to be equipped with 3G/4G radios, whose diffusion will anticipate that of WAVE communication interfaces.



(a) 2-hops.



(b) 5-hops.

Figure 6.15: Spread delay ranked by vehicle density in an area N-hops for the location of the patient zero.

Therefore, that of a complete cellular coverage of WAVE-enabled vehicles seems a reasonable assumption. The problem then becomes that of determining which vehicles to patch. Indeed, simply patching all the vehicles through cellular network downloads can be uselessly expensive. If a rough estimation of the area and time at which the infection started is available, a *smart cellular-based patching* can be adopted, limiting the immunization to vehicles actually interested by the infection.

We thus propose a smart cellular-based patching based on our data-driven model of the worm epidemics. Namely, the model is exploited to determine the region within which the worm may have spread within the time elapsed from the estimated infection

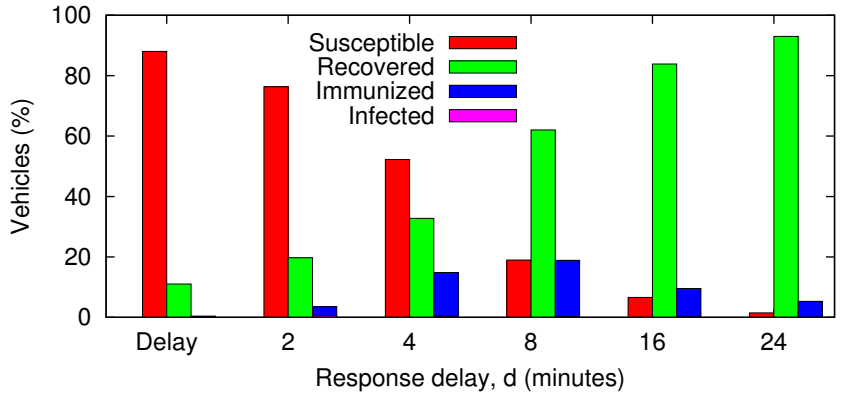


Figure 6.16: Infected, susceptible, healed and immune node ratios.

instant. Then, only vehicles within such a region are immunized through the cellular network.

Fig. 6.16 shows the results of the smart patching in the Canton of Zurich reference scenario, considering that the epidemics starts in the center of Zurich during the peak traffic time. This is a worst-case scenario, since it provides the vehicular malware with maximum network connectivity and thus ideal self-propagation conditions. The response delay d is the estimated time between the patient zero appearance and the instant at which the smart patching is run: clearly, longer response delays imply the infection of larger portions of the road network. For each value of d, along the x axis, we report the number of vehicles belonging to different mutually exclusive categories: *susceptible* nodes were not infected and did not receive the patch, *recovered* nodes were infected and later recovered upon receiving the patch through the cellular network, *immunized* nodes were not infected yet received the patch, and *infected* nodes were infected but did not received the patch. Clearly, the goal of a smart cellular-based patching is to recover all infected nodes, leaving no infected vehicles and reducing the number of unnecessarily immunized nodes to a minimum.

The results show that for a response time of 2 minutes, the model correctly predicts the nodes to be patched, with a negligible number of unnecessarily immunized vehicles. As the response delay increases, the percentage of vehicles infected by the malware grows, leading to the necessity of patching a larger portion of the road traffic. Yet, our model allows to successfully patch all infected vehicles, with a percentage of unnecessarily immunized nodes that stays below 20% even in the worst case, when $d =$ 16 minutes. More importantly, in all cases the percentage of nodes that remain infected

after the smart patching is zero, proving once more the quality of our worm spreading model and its utility towards an efficient containment of malware outbreaks.

6.8 Conclusions

We presented an extensive study of malware spreading in vehicular networks through WAVE V2V communication. Our simulative analysis outlined the high level of danger of vehicular worms, that are shown to be able to spread through very large areas, infecting tens of thousands of vehicles, in a few tens minutes at most. We found that the high mobility of vehicular nodes and the elevate number of short-lived V2V contacts they generate are the key reason behind such a result. We then presented a simple yet very effective data-driven model of the worm propagation process, and leveraged it for the smart patching of infected vehicles through cellular communication.

7

Dissemination of Information in Disruptive/Delay Tolerant Networks under Power Saving Constrains.

7.1 Introduction

In this chapter, the energy saving trade-offs in a DTN (e.g., users with smart-phones) as a function of the *searching* and *sleeping intervals* and as a function of the *node contact duration* is modeled and discussed. The peer-to-peer node contact probability between a mobile node and infrastructure and between two mobile nodes are calculated. These contact probabilities represent, in the first case, the probability that has a node that crosses opportunistically infrastructure during the contact time with the wireless card on, and in the second case, the probability that two nodes that meet also have the wireless interface enabled, and thus can exchange data. We then derive those operating regions in which the nodes can save energy while keeping maximum contact probability. We show that in the case of contacts between mobile nodes and infrastructure there are two possible regions: one with maximum contact probability and other with decreasing contact probability. Furthermore, we show that in the case of contacts between mobile nodes, there are five possible regions: one that offers maximum contact probability and the other four that depending on the on/off intervals and the contact duration will offer

less and less contact probabilities. It is shown that the most conservative strategy to save energy while keeping a contact probability equal to one is to have equal on/off periods lower than the contact interval. Otherwise, the contact probability decreases in order to increase the energy savings.

Furthermore, the impact of peer-to-peer contact probability in the time taken to disseminate a Delay Tolerant Object (DTO) is analyzed. We consider two cases: sparse networks with very few nodes in which Ordinary Differential Equation (ODE) Susceptible-Infection (SI) modeling allows to calculate i) the dissemination time to deliver a DTO to a percentage of users, ii) the mean delivery delay to deliver a DTO to a specific mobile node when the nodes perform power saving strategies and iii) the average energy spent to deliver a DTO. The analysis is extended to sparse networks with higher number of nodes in which spatial correlations dominate. In these cases, the dissemination of information is modeled using the Fisher-Kolmogorov-Petrovsky-Piskonoff (FKPP) Partial Differential Equation (PDE). The FKPP reaction-diffusion equation describes the spatio-temporal evolution of a population in which individuals diffuse with diffusion coefficient D and grow according to a growth function. Klein et al, (10), point that the FKPP model better predicts the dissemination of the information than the ODE model when the number of mobile nodes increases. In this paper, we use the FKPP model with power saving to analyze dissemination delays and the impact of adding sparse infrastructure in the area.

The paper is structured as follows: Section 7.2 reviews the related work. Section 7.3 defines the network model while Section 7.4 derives the peer-to-peer contact probabilities and discuss the trade-offs in power saving in a DTN node. Section 7.5 aims to validate the mathematical model behind the contact probabilities in disruptive/delay tolerant environments. Section 7.6 analyses the impact of the peer-to-peer contact probabilities in the time taken to disseminate delay tolerant data in well-mixed and non-well-mixed scenarios and discusses power savings versus delivery times. Finally, Section 7.7 deals with the conclusions.

7.2 Related Work

Peers in a mobile network alternate between two basic operations: *neighbor discovery* and the *opportunistic transfer of data*. In DTNs, the latter operation has received much

attention as part of routing protocols. However, in a sparse DTN network, searching for other nodes consumes a large percentage of time in comparison to data transfer. Consequently, searching for other nodes becomes the dominant drain on the energy of a battery-powered DTN node. For instance, (129) shows that in some scenarios the use of a 802.11 radio to search for contacts in a DTN requires more than 90% of the total energy simply to find other nodes with which to exchange data.

Previous works have addressed mobile system power management by using 802.11 radios for data transfers and *low-power*, *short-range* radios (e.g., 802.15.4 (130), Bluetooth (131), or CC1000 (132)) for neighbor discovery tasks. This idea of using low-power short-range radios (e.g., CC1000) for neighbor discovery is analyzed in (133). The authors conclude that the addition of a further low power radio introduces a negligible improvement in sparsely mobile DTNs. Via simulation, these authors compare the use of two radios - a low-power radio for discovering contacts and a high-power radio for transmitting data - against a Power Saving Management (PSM) with only a high-power radio. However, the PSM mechanism assumes clock synchronization via GPS. The reason for this is that beacon windows need to be synchronized to start in common discrete intervals. While this solution is appropriate for dense mobile networks, works other than (133), (134) confirm through analytical results that a second short-range radio is indeed inefficient for sparsely populated DTNs. This is because short-range radios miss too many connection opportunities. In this paper we consider a sparse DTN model, and thus focus on a single-radio system.

The use of stationary battery-powered nodes, called throw-boxes, enhances the capacity of DTNs. Banerjee et al, (135), show that without efficient power management, throw-boxes are minimally effective. The authors present a duty-cycled controller for long range radios that predicts when and for how long the mobile node will be in range of the contact duration with the throw-box. The model again needs to beacon position, speed and direction (e.g., using GPS) in order to feed the prediction algorithm. The proposal is tested in the UMass DieselNet Testbed which consists of vehicular mobile nodes. These proposals need GPS data in order to predict contact opportunities or to obtain clock synchronization. In a vehicular network, this assumption does not impose a restriction. But mobile networks consisting of smart-phone users entering closed areas will loose GPS coverage. Moreover, the power consumed in obtaining GPS data is not included in the models.

Finally, Wang et al, (136), investigate the trade-off between the probability of missing a contact and the contact probability frequency in Bluetooth devices. This work shows that in Bluetooth devices the device discovery process consume as much energy as making a phone call and thus achieves the objective of saving energy using adaptive probing mechanisms. Therefore, in their scheme, Bluetooth nodes do not switch off the card and save energy by optimizing discovery frequencies. In our case, we consider Wi-Fi cards in smart-phones that switch the wireless card on and off to save battery, and study the trade-off involved in disabling the wireless card with the contact probability.

7.3 Network Model

Let us assume a network with N wireless mobile nodes and a set of throw-boxes sparsely distributed over a given area. The throw-boxes are back-boned and thus may be considered as a single node (i.e., a node visiting a specific throw-box has access to any data accessible via any other throw-boxes); see Figure 7.1.

A Delay Tolerant Object (DTO) is a set of packets that are disseminated over the area. If nodes u and v meet, u and v get those new DTOs that they do not share. We assume, without loss of generality, that the contact duration between nodes is much longer than the time to exchange a DTO.

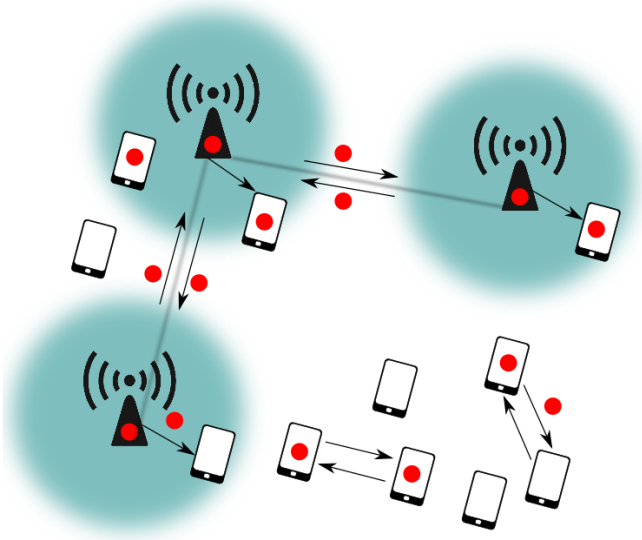


Figure 7.1: Network architecture: DTOs are disseminated via throw-boxes or opportunistically at contact between nodes.

We use the IEEE 802.11 (137) standard in IBSS/Ad-hoc mode as communication technology. The power consumption of IEEE 802.11 cards have been measured in several papers (138), giving for an ORINOCO PC Gold wireless card the following power consumption figures for the different states: {sleeping, idling, receiving, transmission}={60, 805, 950, 1400} mW. Our own measurements on smart-phone devices give results which are in line with the above.

Instead of using the standard procedure for power saving in IBSS/Ad-Hoc mode, which is not supported by most of the devices, we define a Power Saving Management (PSM) mode as follows: Specifically, a node transmitting or receiving a DTO is in the *transmitting or receiving state*. When the node finishes any of these actions, it switches to a PSM mode. The PSM mode consists of switching between two wireless interface states; see Figure 7.2:

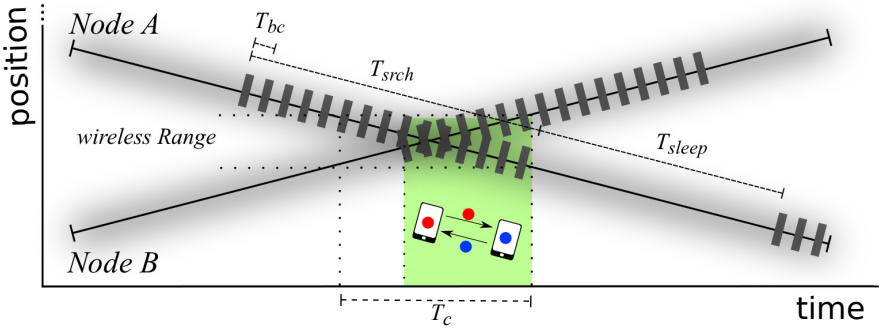
i) sleeping state: a node that is not transmitting or receiving packets, and thus remains in the *sleeping state* during an interval of time equal to T_{sleep}.

ii) search state: a node is in the *idle state* during an interval of time equal to T_{srch}. While the node is in the *idle state*, the node switches periodically (i.e, beacon interval T_{bc}) to the *transmitting state* in order to send a beacon and then returns to the *idle state*.

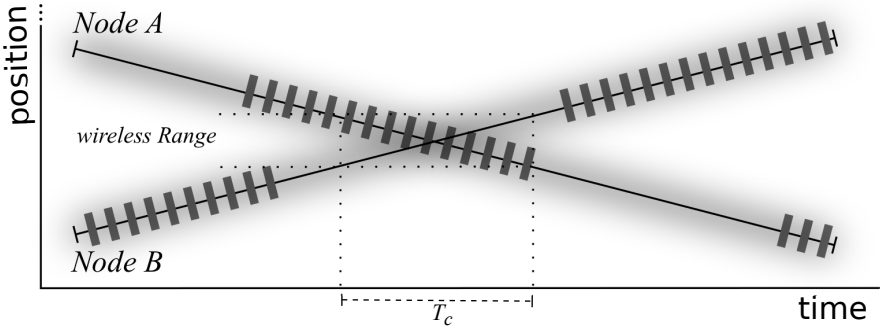
We first note that two nodes would need clock synchronization in order to discover each other if they switched off to the sleeping state after sending their beacons. When a node discovers another node (i.e., listens to its beacons), it initiates a contact exchange as explained in epidemic routing, (139). We secondly note that T=T_{sleep}+T_{srch} and thus the duty-cycle will be of $\frac{T_{srch}}{T}$.

In order to define scenarios for power saving management analysis, we take the example of Nexus One[1] smart-phone with a battery of 1400 mAh and 3.7V that has approximately 250 hours (3G) of stand-by time if the wireless card is off and around 6 hours and a half of lifetime with the wireless card always on. We do not take into account other use of the mobile phone (e.g., phone calls, video, gaming, etc). Four scenarios will be taken as baseline examples:

[1] http://www.google.com/phone/static/en_US-nexusone_tech_specs.html



a) Positive Contact



b) Negative Contact

Figure 7.2: Contact between two nodes with Power Saving Management based on Search/Sleep duty cycling. a) *Positive Contact*: two nodes switch on their Wi-Fi cards while they are in wireless range. b) *Negative Contact*: a node has its Wi-Fi card off while being in wireless range with other node that has its wireless card on (or off).

- **Scen-0:** The smart-phone has the wireless card on all the time. The smart-phone lifetime, following brochure specifications, will be approximately 6 hours and a half.

- **Scen-1:** The smart-phone switches the wireless card on in such a way that the duty-cycle $\frac{T_{srch}}{T} = \frac{1}{3}$. The smart-phone lifetime will be approximately 18 hours.

- **Scen-2:** The smart-phone switches the wireless card on in such a way that the duty-cycle $\frac{T_{srch}}{T} = \frac{1}{6}$. The smart-phone lifetime will be approximately 34 hours.

- **Scen-3:** The smart-phone switches the wireless card on in such a way that the duty-cycle $\frac{T_{srch}}{T} = \frac{1}{9}$. The smart-phone lifetime will be approximately 50 hours.

7.4 Contact Probability

Two nodes that meet and are in the search state have a *positive contact*, Figure 7.2.a), while if they meet and one of them is in the sleep mode, they have a *negative contact*. For example, Figure 7.2.b) represents a scenario in which the nodes meet later than in Figure 7.2.a) as represented by the rectangular areas, where T_C is the contact duration between the two nodes. As it can be observed, in order to have a positive contact, it is necessary that during the contact time, the two nodes switch their wireless card on and their search period overlap. However, there will be a negative contact whenever two nodes meet during a contact period and at least one of them has the wireless card off or the two of them switch the wireless card on but their search period do not overlap.

Let us formalize these ideas. Let T_A and T_B be random variables indicating the time at which mobile nodes A and B switch on from the sleeping state to the search state. T_A and T_B are independent and uniformly distributed with probability density function $f_{T_i}(t)=1/T_i$ for $0 \leq t \leq T_i$ (i={A,B}). Let T_{ctc} be a random variable indicating the time at which a contact would begin if the Wi-Fi cards would always be on and let T_c be the duration of the contact. In sparse networks in which the contact rates are low, the inter-contact time is higher than the period T_i. Thus, when a contact occurs, T_{ctc} is uniformly distributed over the period T and also independent of random variables T_A and T_B. Let us define P_c (contact probability) as the probability that two nodes are in the search state during a contact interval T_c.

$$P_c = \int_{T_A} \int_{T_B} \int_{T_{ctc}} f_{T_A,T_B,T_{ctc}}(\tau_A, \tau_B, \tau_{ctc}) d\tau_A d\tau_B d\tau_{ctc} \tag{7.1}$$

We consider two cases:

i) A mobile node meets opportunistically infrastructure. The mobile node switches on/off its wireless card, while the infrastructure always has its wireless card on.

ii) Two mobile nodes meet opportunistically. Both mobile nodes independently of each other switch on/off their wireless card.

For simplicity, we will consider the case in which $T_A = T_B = T$ (i.e. all nodes has the same on/off period).

7.4.1 Contact Probability between a mobile node and the infrastructure

When the throw-box is not in the transmit/receive state, it will be in the search state looking for mobile nodes that opportunistically contact the throw-box. The simplest case is when $T_c \geq T_{sleep}$, in which the mobile node will have a contact with probability $P_{c_1} = 1$. The case in which $T_c < T_{sleep}$, the contact probability P_{c_1} will depend on whether the mobile node is in the search mode during the contact interval:

$$\begin{aligned} P_{c_1} &= Pr\{(0 \leq T_A \leq T), (T_A - T_c \leq T_{ctc} \leq T_A + T_{srch})\} \\ &= \int_0^T \frac{1}{T} dt \int_{t-T_c}^{t+T_{srch}} \frac{1}{T} d\tau = \frac{T_{srch}+T_c}{T} \end{aligned} \tag{7.2}$$

Table 7.1 summarizes the contact probability for the different situations between mobile nodes and throw-boxes.

Table 7.1: Summary of contact probability, P_{c_1}, between mobile nodes and throw-boxes.

$P_{c_1} = \frac{T_{srch}+T_c}{T}$	$T_c < T_{sleep}$
$P_{c_1} = 1$	$T_c \geq T_{sleep}$

7.4.2 Contact Probability between two mobile nodes

There will be a set of different possibilities for calculating the contact probability, P_{c_2}, depending on the lengths of the sleeping state, T_{sleep}, the search state, T_{srch}, and the contact duration, T_c. There are five possible cases, which are summarized in Table 7.2.

In order to calculate the different cases in table 7.2, let us consider the following situations:

- When $T_c \leq T_{sleep}$ the contact probability can be calculated as $P_{c_2} = 2 \cdot Pr\{(0 \leq T_A \leq T), (T_A - T_{srch} \leq T_B \leq T_A), (T_A - T_c \leq T_{ctc} \leq T_B + T_{srch})\}$:

$$\begin{aligned} P_{c_2} &= \frac{2}{T^3} \cdot \int_0^T dt \int_{t-T_{srch}}^{t} ds \int_{t-T_c}^{s+T_{srch}} d\tau \\ &= \frac{(T_{srch}+T_c)^2 - T_c^2}{T^2} \end{aligned} \tag{7.3}$$

- When $T_{sleep} \leq T_c \leq 2T_{sleep}$ and $T_{sleep} \leq T_{srch}$, the contact probability can be calculated as $P_{c_2} = Pr\{(0 \leq T_A \leq T), (T_A - T_{srch} \leq T_B \leq T_A - T_{sleep})\} + Pr\{(0 \leq$

$T_A \leq T), T_A - (T_c - T_{sleep}) \leq T_B \leq T_A + (T_c - T_{sleep})\} + 2Pr\{(0 \leq T_A \leq T), (T_A - T_{sleep} \leq T_B \leq T_A - (T_c - T_{sleep}), (T_A - T_c \leq T_{ctc} \leq T_B + T_{srch})\}$:

$$\begin{aligned} P_{c_2} = & \frac{1}{T^2} \cdot \int_0^T dt[\int_{t-T_{srch}}^{t-T_{sleep}} ds + \int_{t-T_{sleep}}^{t+(T_c-T_{sleep})} ds] \\ & + \frac{2}{T^3} \int_0^T dt \int_{t-T_{sleep}}^{t-(T_c-T_{sleep})} ds \int_{t-T_c}^{s+T_{srch}} d\tau \\ & = 1 - \frac{(T_c-2T_{sleep})^2}{T^2} \end{aligned} \tag{7.4}$$

- When $T_{srch} \leq T_{sleep} \leq T_c \leq T$, the contact probability can be calculated as $P_{c_2} = 2 \cdot [Pr\{(0 \leq T_A \leq T), (T_A - T_{srch} \leq T_B \leq T_A - (T_c - T_{sleep})), (T_A - T_c \leq T_{ctc} \leq T_B + T_{srch})\} + Pr\{(0 \leq T_A \leq T), (T_A - (T_c - T_{sleep}) \leq T_B \leq T_A)\}]$:

$$\begin{aligned} P_{c_2} & = \frac{2}{T^3} \cdot [\int_0^T dt \int_{t-T_{srch}}^{t-(T_c-T_{sleep})} ds \int_{t-T_c}^{s+T_{srch}} d\tau \\ & + \int_0^T dt \int_{t-(T_c-T_{sleep})}^{t} ds] \\ & = \frac{(T_{srch}+T_c)^2+(T_c-T_{sleep})^2-T_c^2}{T^2} \end{aligned} \tag{7.5}$$

- When $T \leq T_c$ and $T_{srch} \leq T_{sleep}$, the contact probability can be calculated as $P_{c_2} = 2 \cdot Pr\{(0 \leq T_A \leq T), (T_A - T_{srch} \leq T_B \leq T_A)\}$:

$$P_{c_2} = 2 \cdot \int_0^T \frac{1}{T} dt \int_{t-T_{srch}}^{t+T_{srch}} \frac{1}{T} ds = \frac{2T_{srch}}{T} \tag{7.6}$$

- Finally, $P_{c_2} = 1$ when $2T_{sleep} \leq T_c$ and $T_{sleep} \leq T_{srch}$.

Table 7.2: Summary of contact probability, P_{c_2}, between mobile nodes as a function of parameters T_{srch}, T_{sleep} and T_c.

$P_{c_2} = \frac{(T_{srch}+T_c)^2-T_c^2}{T^2}$	$T_c \leq T_{sleep}$
$P_{c_2} = 1 - \frac{(T_c-2T_{sleep})^2}{T^2}$	$T_{sleep} \leq T_c \leq 2T_{sleep}$ and $T_{sleep} \leq T_{srch}$
$P_{c_2} = \frac{(T_{srch}+T_c)^2-(T_c-T_{sleep})^2-T_c^2}{T^2}$	$T_{srch} \leq T_{sleep} \leq T_c \leq T$
$P_{c_2} = \frac{2T_{srch}}{T}$	$T \leq T_c$ and $T_{srch} \leq T_{sleep}$
$P_{c_2} = 1$	$2T_{sleep} \leq T_c$ and $T_{sleep} \leq T_{srch}$

7.4.3 Trade-offs between contact probability and power saving

Let us define the following parameters that describe the aggressiveness of the different states, one that measures the powersaving gain, while the other measures the relation between the contact duration and the T_{srch}: $b_1 = \frac{T_{sleep}}{T_{srch}}$ and $b_2 = \frac{T_c}{T_{srch}}$. Note that $T = T_{srch} + T_{sleep} = T_{srch}(1 + b_1)$. Thus, the **duty-cycle** $\frac{T_{srch}}{T} = \frac{1}{1+b_1}$ and parameter b_1 is related to how much battery the node will save. Let us assume that a mobile node with the wireless interface always on has a lifetime of L_t seconds. A power saving in which the node performs a duty-cycle of $\frac{T_{srch}}{T}$, the node would increase its lifetime by:

$$L_{dt} = \frac{T}{T_{srch}} L_t = (b_1 + 1) L_t \tag{7.7}$$

As equation (7.7) shows, the lifetime increment ΔL_t is proportional to parameter b_1 (i.e., $\Delta L_t = L_t - L_{dt} = b_1 L_t$). Thus, $b_1 = 0$ (i.e. duty-cycle=1) means that the node always has the wireless interface on, while $b_1 = \infty$ (i.e. duty-cycle=0) means that the node always has the wireless interface off. On the other hand, large values of b_2 mean that the contact interval is larger than the search interval, while values of $b_2 < 1$ mean that the contact interval is lower than the search interval. Parameters b_1 and b_2 allow us to analyze the impact of T_{srch}, T_{sleep} and T_c without specifying absolute values for these time intervals. Tables 7.3 and 7.4 summarize contact probabilities P_{c_1} and P_{c_2}, respectively, as a function of b_1 and b_2. Finally, Figure 7.3 shows the different regions for contacts between two mobile nodes depending on parameters b_1 and b_2.

Table 7.3: Summary of contact probability, P_{c_1}, between mobile nodes and throw-box.

Region	Contact Probability	Intervals
A_1	$P_{c_1} = \frac{1+b_2}{1+b_1}$	$b_2 < b_1$
B_1	$P_{c_1} = 1$	$b_2 \geq b_1$

Let us begin by representing P_{c_1} graphically. Figure 7.4 depicts the two regions identified in Table 7.3 and shows that if $b_2 \geq b_1$ it does not matter what the size of T_{srch} is: in this case, one would choose the minimum possible T_{srch} (i.e. the beacon interval) in order to save as much energy as possible, while keeping the maximum contact probability, since immediately after switching on the wireless card the node will detect the throw-box. Note, that the optimal operation points are those on the straight line that divides the two regions (i.e. $b_1 = b_2$), because they correspond to the

Table 7.4: Summary of contact probability, P_{c_2}, between mobile nodes.

Region	Contact Probability	Intervals
A_2	$P_{c_2} = \frac{1+2b_2}{(1+b_1)^2}$	$b_2 \leq b_1$
B_2	$P_{c_2} = 1 - \frac{(b_2-2b_1)^2}{(1+b_1)^2}$	$b_1 \leq b_2 \leq 2b_1$ and $b_1 \leq 1$
C_2	$P_{c_2} = \frac{(1+b_2)^2-(b_2-b_1)^2-b_2^2}{(1+b_1)^2}$	$1 \leq b_1 \leq b_2 \leq 1+b_1$
D_2	$P_{c_2} = \frac{2}{(1+b_1)}$	$1+b_1 \leq b_2$ and $1 \leq b_1$
E_2	$P_{c_2} = 1$	$2b_1 \leq b_2$ and $b_1 \leq 1$

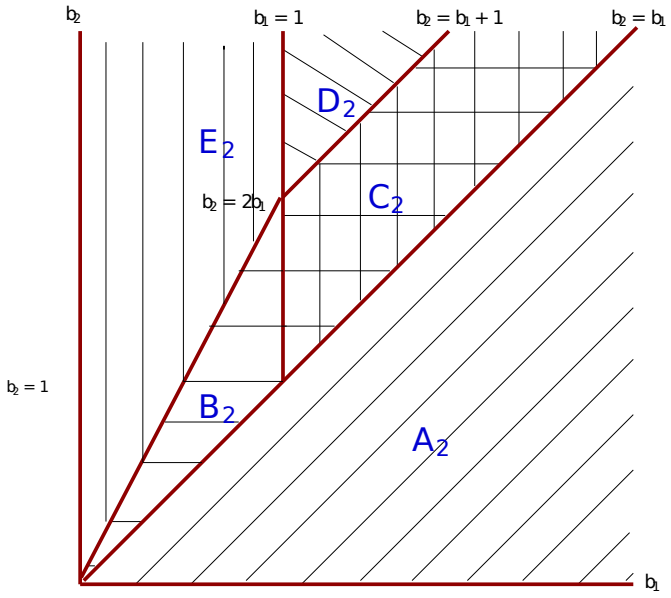


Figure 7.3: Contact areas between two mobile nodes.

maximum b_1 (maximum energy saving) allowed while keeping $P_{c_1} = 1$. This is what wireless cards actually do for detecting access points: operate with low b_1 while keeping the T_{sleep} so low, that always $b_2 \geq b_1$. On the other hand, if $b_2 < b_1$, the system will inevitably suffer from lost contacts. This effect can be minimized when b_2 approaches b_1. Asymptotically, making T_{srch} very large would allow this objective to be reached because both b_1 and b_2 would tend to 0. Note, however, that this is not a practical approach since it would imply a large waste of energy. The system must decrease b_1 by decreasing its T_{sleep} as near as possible to T_c to move to the operating region with $P_{c_1} = 1$. So an inherent trade-off exists between contact probability and energy saving.

Let us now turn our attention to Figure 7.5, which represents P_{c_2} as defined in

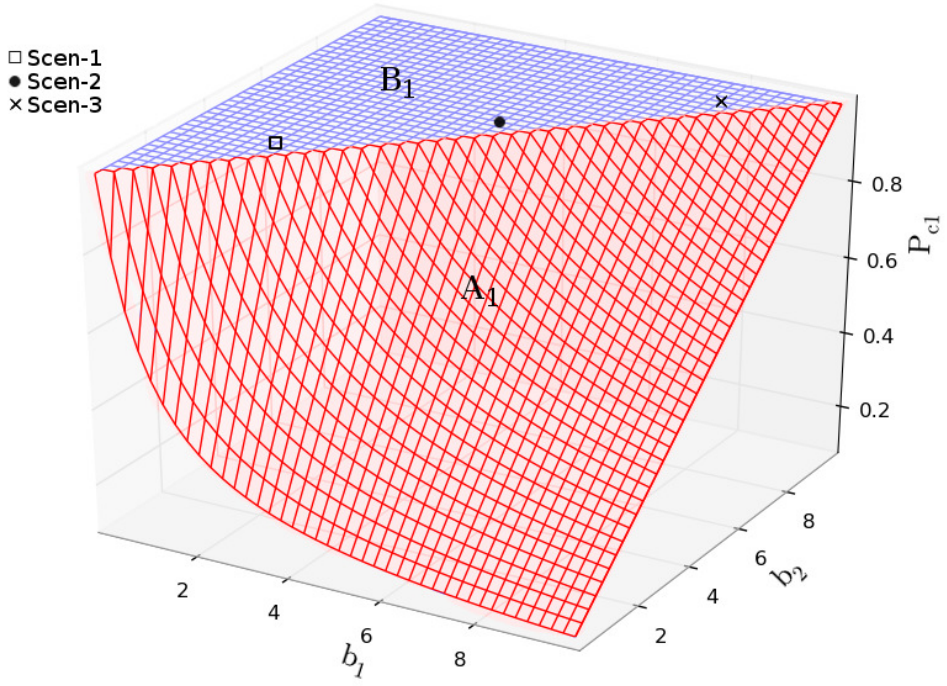


Figure 7.4: Contact probability, P_{c_1}, between mobile nodes and throw-boxes.

Table 7.4. The 5 regions of operation can be identified as depicted in Figure 7.3. Region A_2 is the worst in terms of contact probability. In this region, for a target b_1, low values of b_2 yield low contact probabilities. For a given duration contact T_c, an increase of b_2 implies a decrease in T_{srch}. However, this step would move (increase) b_1 to a new point with lower P_{c_2}. Thus, in order to keep the same P_{c_2}, T_{sleep} should be proportionally decreased. The best contact probability is achieved when b_2 approaches b_1, or in other words, when T_{sleep} approaches T_c. In this case, the contact probability P_{c_2} will be that between A_2 and B_2 and C_2 frontiers. In order to increase b_1, and thus save more batteries, T_{srch} must decrease or T_{sleep} must increase at the cost of decreasing P_{c_2}. The reason for this behavior is quite evident: T_{sleep} is larger than T_c and the larger T_{sleep} becomes, the larger the probability of missing more contacts.

Two regions that represent higher values of P_{c_2} are B_2 and C_2. B_2 presents the problem that $b_1 \leq 1$ which implies that the sleep period must be shorter than the search period (thus limiting the achievable energy savings), so it is a region completely without interest: the same energy savings can be achieved at region E_2 but with $P_{c_2} = 1$. So

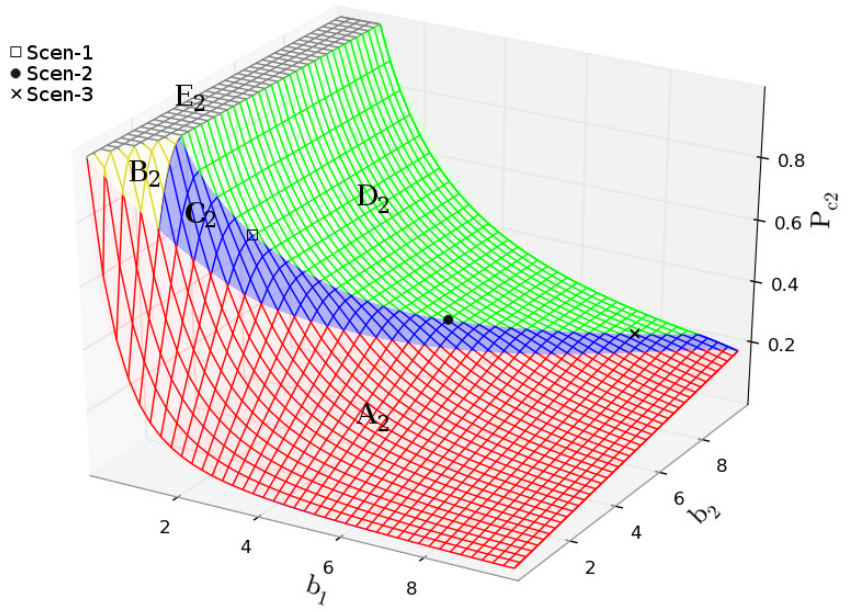


Figure 7.5: Contact probability, P_{c_2}, between mobile nodes.

a node operating in this region should increase b_2 (i.e. reduce T_{srch}) while keeping b_1 (i.e. reducing T_{sleep} accordingly), in order to move to operating region E_2. Note also that $b_1 = 1$ is the best option in terms of P_{c_2}: the combination of both strategies will conduct the system to an operating point with $P_{c_2} = 1$, which is the intersection between regions B_2, C_2, D_2 and E_2 and corresponds to $b_1 = 1$ and $b_2 = 2$ (i.e., $T = T_c$ and $T_{srch} = T_{sleep} = T_c/2$). At this point, $P_{c_2} = 1$, while $b_1 = 1$, which implies a duty-cycle of 1/2: duplicates the lifetime when compared with a card always on. If a node wants to increase more its lifetime, he has to decrease its duty-cycle in such a way that $b_1 > 1$. That means moving its operating point to regions A_2, C_2 or D_2.

Note that ($b_1 = 1$, $b_2 = 2$) is not the only point that allows for $P_{c_2} = 1$: the whole frontier between regions D_2 and E_2 allows for that, but between all the points in the frontier, this is the one with lower b_2 (i.e. higher T_{srch} for a fixed T_c). Increasing b_2 would imply decreasing T_{srch}; but in order to keep b_1 fixed at one, $T_{sleep} = T_{srch}$. The result implies switching the wireless card on/off many times. This is important from a practical point of view, because the process of switching a wireless card on/off is energy - and time - consuming, so a system with larger T is more desirable. Operating in other points of region E_2 implies that $b_1 \leq 1$ and thus less battery is saved without

any gain in P_{c_2}. Moving out from region E_2, region D_2 is the more favorable case from the contact probability point of view for two reasons: it allows for relatively high values of P_{c_2} while introducing higher energy savings. Note that for a given b_1, P_{c_2} remains constant for any b_2. This implies that by keeping the ratio $\frac{T_{sleep}}{T_{srch}} = b_1 > 1$, T_{srch} is limited by T_c (i.e. $1 + b_1 \leq b_2$ implies $T \leq T_c$ and $T_{srch} \leq T_c/(1 + b_1)$).

This section can be summarized with the following conclusions: given that nodes have a percentage of contacts with duration higher than a certain value T_c, then, designing an on/off period with values $T_{srch} = T_{sleep} = T_c/2$ assures a battery saving of around 50%, since nodes would have these percentage of contacts operating over the frontier E_2-D_2. The remaining contacts would occur with a probability according to the region in which they take place. Higher battery savings may be obtained if $T \leq T_c$ and $T_{srch} \leq T_{sleep}$ (region D_2) at the cost of reducing peer-to-peer contact probabilities. Regions A_2 and C_2 also allow energy savings, but with worse contact probabilities. We will come back to this issue in section 7.6.

7.5 Model Validation

We considered using real mobility traces or contact logs as inputs for simulating the previous analysis. For example (140), (141) are trace based studies freely available for download and use. However, this choice was discarded because of the large granularity they used for the traces: the node starts a device discovery at each time step and logs the nodes under coverage. As it will be shown later, assuming a wireless range of 100m, 20% of the contact durations take less than 30s. So, any log with a higher granularity may miss some of the contacts. If we change the wireless range to the approximately 10m of Bluetooth, the amount of contacts that may be missed while using such granularity can be really meaningful for our study. Granularity also affects to the observed contact duration. A contact will last a discrete number of time-steps while these can be one order of magnitude longer than the real duration. In (140), the Bluetooth antennas of iMotes are used to log contacts with a granularity of 120s, while (141) uses Bluetooth antennas of mobile phones to log contacts with a granularity of 300s, while we observed a meaningful amount of contact durations that were shorter than 30s. Unless power limitations are solved, with low power consumption antennas, or higher capacity batteries, thus allowing devices always to remain on, real test-beds cannot be

improved to log accurately all the contacts among nodes with finer granularity. These facts encouraged us to use a mobility simulator in order to validate the mathematical model.

In order to evaluate the model, let us consider a urban scenario where pedestrian nodes equipped with wireless devices switch between the sleeping state and the search state with the given durations T_{sleep}, T_{srch}. Our aim is to compare the contact ratio obtained in the simulations with the results obtained from the formulas. The pedestrian mobility traces employed were obtained with the UDel Models (42). The UDel Mobility Model considers both micro and macro mobility, and is based on (i) real surveys collecting detailed information about how people spend their time, (ii) a task model that focuses on the mobility of people inside buildings, and (iii) an agent model that determines how mobile nodes interact with each other (e.g., each node has a day-long realistic behavior where they wake-up, go to work, have lunch, etc., while avoiding collisions and overtaking slower pedestrians, or waiting at traffic lights). The scenario used is $Chicago9Blk$, included in the default UDel Models Data Set. It consists of a city segment of 400x400m, similar to a "Manhattan city model" of 4 by 4 streets, where 100 pedestrian nodes spend 12 hours, starting at 8.00AM. Any other vehicular, UAV or static node are not considered. 20 instances of this scenario are obtained using different initial random seeds with the default Udel Mobility Parameters for the pause time, speed, lane-changing, distance-speed relationship, etc.

To avoid any kind of synchronization among nodes, the simulator initializes the nodes, giving a random start over the period T for each node. Then, the mobility traces are used to obtain the coordinates of each node. The simulator considers the position and the *sleeping/search* state of the nodes in order to log the positive contacts. For the scenario $Chicago9Blk$, we have obtained a contact rate, β, between mobile nodes equal to 0.0071 contacts/s. We note that the positive contact rate between mobile nodes will be equal to $\beta' = P_{c_2}\beta$. The two first steps consist of obtaining the CDF of the contact duration and validating the Peer-to-peer Contact probability, P_{c_2}, shown in Figure 7.5.

The CDF of the contact duration is shown in Figure 7.6. One may observe how approximately 70% of the contacts have contact durations lasting less than 500 seconds, while the other 30% of the contacts have contact durations in excess of 500 seconds. Figure 7.6 also zooms contact durations of less than 500 seconds. One may observe

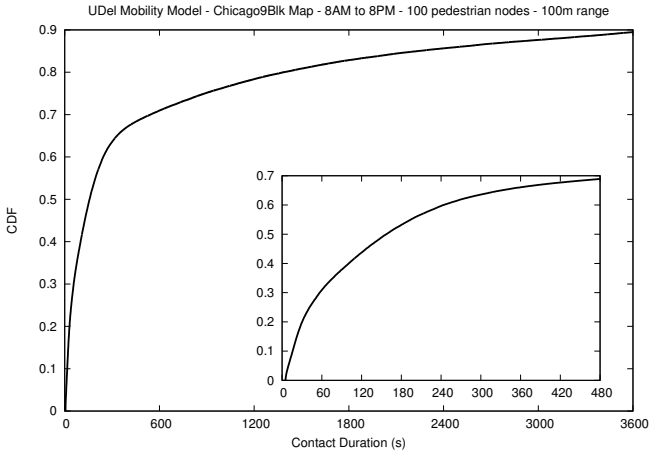


Figure 7.6: CDF of the contact duration

that only 20% last less than 30 seconds, and only 30% of the contacts have contact durations less than one minute. These values provide us with a hint about the amount of time nodes could fix b_1 and b_2 and will be used in the next section.

In order to validate the peer-to-peer contact probability, P_{c_2}, Figure 7.7 shows the simulated results against the contact probabilities as obtained in Table 7.4 and Figure 7.5. For the sake of clarity, instead of comparing the 3D plots, cross sections for several b_1 values are drawn as a function of b_2. The simulated values are obtained with a confidence interval of 99%. As it can be observed, the simulations perfectly match the analytical values.

Figure 7.8 shows the peer-to-peer contact probability, P_{c_2}, as a function of T_{sleep} for different values of $T_{srch} = \{5, 30, 60, 120\}$ seconds, obtained with the simulator. Two main points can be observed: the first one is that for a fixed T_{srch}, increasing T_{sleep} implies a low duty cycle and the peer-to-peer contact probability decreases very fast. The second fact is that for the same ratio $b_1 = T_{sleep}/T_{srch}$ (e.g., see the points marked with a box and a circle corresponding to $b_1 = 2$ and $b_1 = 3$), having longer T_{srch} periods imply lower peer-to-peer contact probabilities. Finally, the curves show how the simulation, using a confidence interval of 99% matches the mathematical analysis quite well.

This is further proved in Figure 7.9, where the contact probability is shown as a function of $T = T_{srch} + T_{sleep}$ for different values of $b_1 = \{0.5, 1, 2\}$. For $b_1 = 0.5$ and 1 both simulations and the mathematical model follow the same line, while for $b_1 = 2$

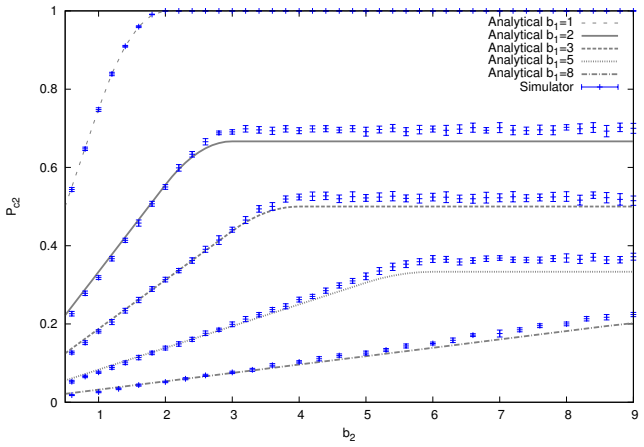


Figure 7.7: Contact probability, P_{c_2}, between mobile nodes analytical values against simulated ones with a confidence interval of 99%.

they conform to the same behavior. However, for small values of T the difference is bigger. The figure shows a slow decay in the contact probability as a function of the total period T. These figures confirm those results obtained in the previous section, where it was shown that in order to have high energy savings the contact probability rapidly decreases.

7.6 Data Dissemination in Sparse Networks

Let us assume that x% of the contacts has contact durations higher than T_C seconds and let us consider a scenario with a duty-cycle $\frac{T_{srch}}{T}$ such that $b_1 > 1$. Then, T_{sleep}=$b_1 T_{srch}$ and $T = (b_1 + 1)T_{srch}$. Since $b_1 > 1$, the region that have better contact probability is region D_2 with conditions $b_2 \geq 1 + b_1$ (i.e., $T \leq T_C$) and $b_1 \geq 1$ (i.e., $T_{srch} \leq T_{sleep}$). Thus, fixing $T_{srch} = T_c/(1+b_1)$ assures that x% of the contacts have contact probability $P_{c_2} = \frac{2}{(1+b_1)}$. It is important to stress that T_C is not controlled, it is to say, those contacts that have a duration less than the target T_C will lie in operating regions C_2 and A_2, yielding lower contact probabilities. However, a good choice of parameters T_{srch} and T_{sleep} will produce that many of the contacts will operate in regions E_2 or D_2.

For instance, using Figure 7.6, 80% of the contacts have contact durations higher than 30 seconds. Table 7.5 shows the design parameters for the scenarios defined in

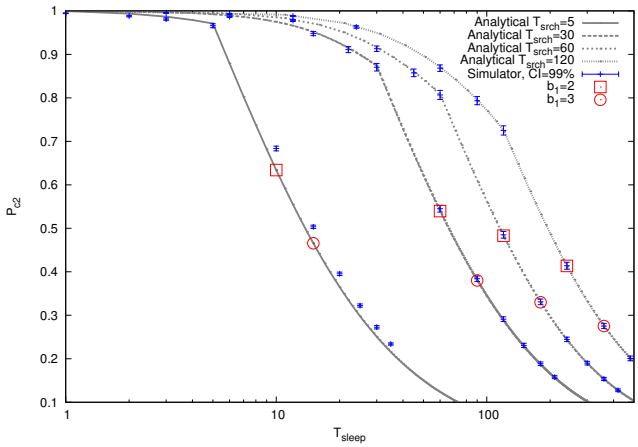


Figure 7.8: Peer-to-peer contact probability as a function of T_{sleep}

section 7.3 having contacts with $T_c \geq 30$s as target.

Table 7.5: Data parameters for Scen-0 to Scen-3 with $T_c \geq 30$s and $\beta = 3.84 \times 10^{-4}$.

Scenario	b_1	b_2	P_{c_1}	P_{c_2}	$\beta \cdot P_{c_2}$ ($\times 10^{-4}$)	T_{srch} (s)	T_{sleep} (s)
Scen-0	0	0	1	1	3.84	∞	0
Scen-1	2	3	1	0.66	2.53	10	20
Scen-2	5	6	1	0.33	1.26	5	25
Scen-3	8	9	1	0.22	0.84	3.33	26.6

As an example of use, we will consider the dissemination of a DTO in a closed area and will use epidemic routing with the aim of characterizing the impact of lower contact probabilities in the trade off between power saving and dissemination time. We will use the ODE model defined in (18), subsection 7.6.1, and a simple mobility model such as random direction mobility model to show these trade-offs. As it is shown by Klein et al in (142), epidemic analysis using the ODE model of (18) is accurate in the well-mixed scaling law - i.e., in very sparse scenarios. For denser scenarios a PDE modeling using diffusion is needed, subsection 7.6.2.

7.6.1 Very sparse networks: the well-mixed case

Let us consider the time origin as the time at which a node creates a DTO, and let $s(t)$ be the proportion of nodes that have a copy of the DTO at time t. The rate at which

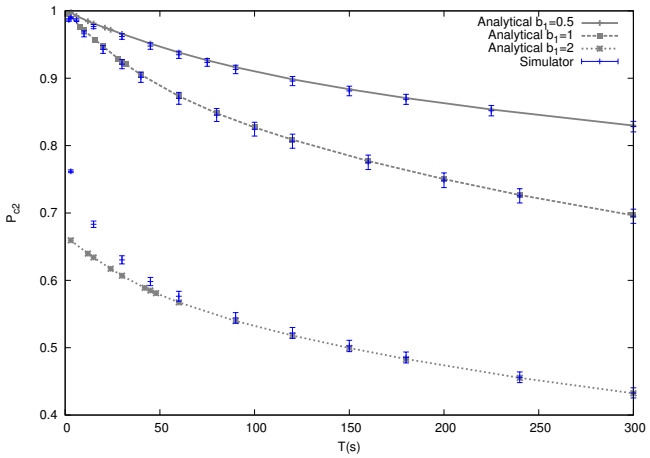


Figure 7.9: Peer-to-peer contact probability as a function of the period T

the fraction of nodes that have received a copy of the DTO changes, (18), is given by the ordinary differential equation (ODE):

$$\begin{aligned} \frac{d}{dt}s(t) &= (P_{c_1}\alpha + P_{c_2}\beta s(t))(1-s(t)) \\ &= -P_{c_2}\beta s^2(t) + (P_{c_2}\beta - P_{c_1}\alpha)s(t) + P_{c_1}\alpha \end{aligned} \tag{7.8}$$

where α is the contact rate between a mobile node and the infrastructure, β is the contact rate between mobile nodes, and P_{c_1} and P_{c_2} are the contact probabilities between a node and a throw-box and between mobile nodes when nodes perform power saving management. This type of equation corresponds, (18), to the fluid limit of a Markov model as N increases. Note that eq. (7.8) is a Riccati ordinary differential equation that can be solved by substituting $s(t) = \frac{1}{P_{c_2}\beta}(\frac{v'(t)}{v(t)})$. Thus, solving eq. (7.8), the proportion $s(t)$ of nodes that have a copy of the DTO at time t is given by:

$$s(t) = \frac{P_{c_1}\alpha + P_{c_2}\beta s(0) - P_{c_1}\alpha(1-s(0))e^{-(P_{c_1}\alpha+P_{c_2}\beta)t}}{P_{c_1}\alpha + P_{c_2}\beta s(0) + P_{c_2}\beta(1-s(0))e^{-(P_{c_1}\alpha+P_{c_2}\beta)t}} \tag{7.9}$$

where $s(0)=\frac{1}{N}$ is the initial proportion of nodes with a DTO copy. Letting α=0, eq.(7.9) gives us the proportion of nodes using power saving that receive the DTO without infrastructure and letting β=0, eq. (7.9) gives us the proportion of nodes that receive the DTO only contacting infrastructure.

Let us define the *dissemination time*, T_D as the time needed to disseminate a DTO to a percentage $s(T_D)$ of users. Thus, the dissemination time T_D can be computed from eq. (7.9) as:

$$T_D = \frac{1}{(P_{c_1}\alpha + P_{c_2}\beta)} ln[\frac{(1-s(0))(P_{c_1}\alpha + P_{c_2}\beta s(T_D))}{(1-s(T_D))(P_{c_1}\alpha + P_{c_2}\beta s(0))}] \quad (7.10)$$

with $s(T_D) \in [\frac{1}{N}, \frac{N-1}{N}]$. Table 7.6 summarizes the dissemination times for the general case (i.e., $\alpha \neq 0$, $\beta \neq 0$) and for the cases in which there is no infrastructure (i.e., $\alpha = 0$, $\beta \neq 0$) and there is only infrastructure (i.e., $\alpha \neq 0$, $\beta = 0$).

Table 7.6: Summary of dissemination times,T_D, with and without infrastructure.

$T_D = \frac{1}{(P_{c_1}\alpha+P_{c_2}\beta)} ln[\frac{(1-s(0))(P_{c_1}\alpha+P_{c_2}\beta s(T_D))}{(1-s(T_D))(P_{c_1}\alpha+P_{c_2}\beta s(0))}]$	$\alpha \neq 0$, $\beta \neq 0$
$T_D = \frac{1}{(P_{c_1}\alpha)} ln(\frac{(1-s(0))}{(1-s(T_D))})$	$\alpha \neq 0$, $\beta = 0$
$T_D = \frac{1}{(P_{c_2}\beta)} ln(\frac{s(T_D)}{(1-s(T_D))} \frac{(1-s(0))}{s(0)})$	$\alpha = 0$, $\beta \neq 0$

Figure 7.10 shows the dissemination times needed to reach different percentage $s(T_D)$ of users for several values of α (no infrastructure, and average contact rates of 10^{-3} and 10^{-4} seconds with the infrastructure) and for each scenario. The area is the same as that one used by Klein et al in (142) for a well-mixed scaling law: an area of 16 square Km and N=25 users using random direction with reflection. This scenario has been validated in (142) as representative of a very sparse scenario and is well modeled via ODE analysis. We have chosen a radio range of 100 meters and users move at 1 m/s. The resulting average contact rate is $\beta = 3.84 \times 10^{-4}$. From the figure it is clear that it is extremely costly to reach a high percentage $s(T_D)$ of nodes when there is no infrastructure. This cost is worse in scen-3, with a lower contact probability. As an example, targeting a percentage of users of $s(T_D)$=80%, the dissemination time without infrastructure is approximately of 15 hours in Scen-3, 10 hours in Scen-2, 5 hours in Scen-1 and 3 hours 20 minutes in Scen-0. Adding infrastructure assures that most of the DTOs are obtained when the nodes cross the infrastructure. One may observe that operating in region D_2 implies that $b_2 \geq b_1 + 1$, and thus, nodes crossing throw-boxes coverage operate in region A (i.e., $P_{c_1} = 1$). For example, the dissemination times decrease to values ranging between 22 to 25 minutes for the four scenarios when $\alpha = 10^{-3}$ (i.e., crossing the infrastructure every 16.6 minutes). Furthermore, since contacts operate in region A with the infrastructure, almost all deliveries are done by the throw-boxes and not by the peer-to-peer contacts.

Let us calculate how much energy a node spends until a node receives a DTO. We call this parameter *Energy-to-Deliver a DTO*. The *Mean Delivery Delay*, MDD,

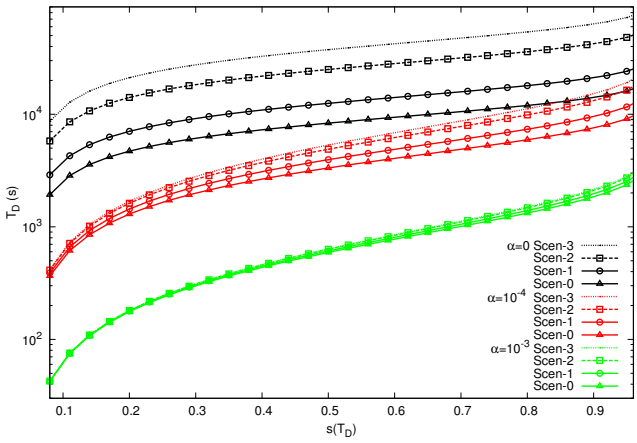


Figure 7.10: Dissemination time, T_D, as a function of the percentage of users, $s(T_D)$ for the three scenarios. $\beta = 3.84 \times 10^{-4}$ and $N = 25$ users.

is the expected delay between the DTO generation at the source and the delivery to a destination. The MDD can be calculated averaging the T_D over the $s(T_D)$. Then, with a duty cycle strategy and knowing that the duty cycle is $\frac{T_{on}}{T} = \frac{1}{(1+b_1)}$, the energy consumed for receiving a DTO will be:

$$E = MDD[(\frac{1}{1+b_1})P_{srch} + (\frac{b_1}{1+b_1})P_{sleep}] \tag{7.11}$$

where P_{srch} and P_{sleep} are the power consumptions in each state. For example, if $\alpha = 0$ - without infrastructure - the MDD is given by MDD=$\frac{ln(N-1)}{\beta P_{c_2}}$. In this case, Scenario 0 with a $b_1 = 0$, $P_{c_2} = 1$ and a $\beta = 3.84 \times 10^{-4}$ has a MDD=8.341×10^3 s. For the reference smart-phone - a Nexus one - with a battery of 1400 mAh and 3.7V, the energy consumed during this time is of $E = 6.673 \times 10^3$ J. Assuming that for $b_1 \leq 1$ nodes operate in region E_2 and for $b_1 \geq 1$ nodes operate in region D_2, the energy consumed for receiving a DTO will be:

$$E = \begin{cases} \frac{ln(N-1)}{\beta(1+b_1)}[P_{srch} + b_1 P_{sleep}] & \text{if } b_1 \leq 1 \quad 2b_1 \leq b_2 \\ \frac{ln(N-1)}{2\beta}[P_{srch} + b_1 P_{sleep}] & \text{if } b_1 \geq 1 \quad 1 + b_1 \leq b_2. \end{cases} \tag{7.12}$$

Figure 7.11 shows the trade off between the energy needed to receive a DTO and the duty cycle $\frac{T_{on}}{T} = \frac{1}{(1+b_1)}$. From the figure, it can be observed that there are two zones: the first zone corresponding to $b_1 \leq 1$ - duty cycles between 1 and 0.5 - is the result of $P_{c_2} = 1$. Since the MDD is constant in this interval, the optimal strategy is

to have a duty cycle of 0.5 (i.e., $b_1 = 1$) with a minimum average energy consumption of $\frac{ln(N-1)}{2\beta}[P_{srch} + P_{sleep}]$. The second region corresponds to lower duty cycles (i.e., higher b_1). However, since $P_{c_2} < 1$, MDD will be higher and more on/off periods will be needed in order to obtain the DTO. Designing very low duty cycle strategies is impractical since the MDD is too high and the average energy consumption to obtain a DTO will increase due to the number of on/off periods needed to reach this MDD time.

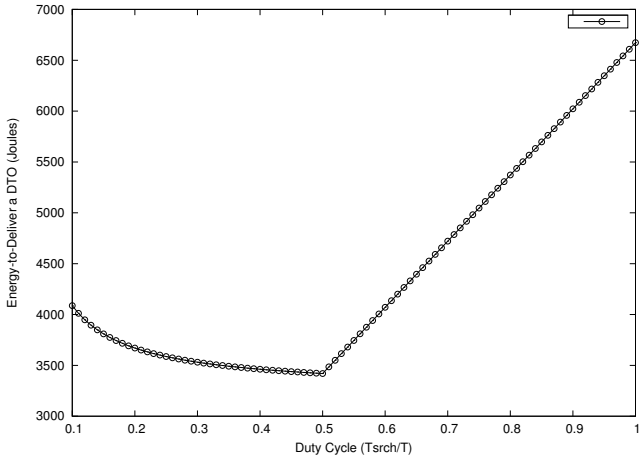


Figure 7.11: Energy needed to receive a DTO versus the duty cycle.

7.6.2 Sparse networks: the non well-mixed case

In denser regimes, known as non well-mixed scenarios, (142), spatial correlations dominate. Thus, in these cases, the dissemination of information is modeled using the Fisher-Kolmogoroff-Petrovsky-Piscounoff (FKPP) Partial Differential Equation (PDE):

$$\begin{aligned} \frac{d}{dt}s(x,t) &= D\Delta s(x,t) + F(s(x,t)) = \\ &= D(\frac{d^2}{dx_1^2} + \frac{d^2}{dx_2^2})s(x,t) + F(s(x,t)) \end{aligned} \tag{7.13}$$

The FKPP reaction-diffusion equation describes the spatio-temporal evolution of a population in which individuals diffuse with diffusion coefficient D and grow according to a growth function $F(s(x,t))$. Here $s(x,t)$ is the proportion of nodes that have a copy of the DTO at time t at position $x = (x_1, x_2)$. Of course, the growth function in our case, without considering infrastructure, would be in the form:

$$F(s(x,t)) = P_{c_2}\beta s(x,t))(1 - s(x,t)) \tag{7.14}$$

The solution of this PDE is a waveform of form s(x,t)=Θ_v(x-vt). Kolmogorov et al, in 1937, proved that the minimal front speed is $v_{min} = 2\sqrt{DF'(0)}$. It can be easily shown that for a node with radio coverage R that initiates the infection in an area of size A with population N, $F'(0)$=$P_{c_2}\beta$ and β=$2RV_eN/A = 2RV_e\lambda_N$, where $\lambda_N = N/A$ is the node density in the given area. Then, the wave speed is $v_{min} = 2\sqrt{P_{c_2}\beta D} = 2\sqrt{2P_{c_2}RV_eD\lambda_N}$.

We point out that the growth function $F(s(x,t))$ does not include the contact rate with infrastructure. This is due to the fact that in the ODE SI (Susceptible-Infected) model each access point is considered as a node that participates in the infection. Its location was not taken into account and only the contact rate between mobile users and access points were considered. However, in the FKPP PDE model, the source location of the infection plays a key role since it fixes the initial point at which the wave front initiates its propagation. Having several access points means that there are several sources or focuses and their locations have to be taken into account. We will come to this point later.

In order to calculate the wave speed v_{min}, we need to obtain the diffusion coefficient D. This parameter is related to the mean square displacement (MSD) defined as an average square distance traveled by a mobile node over time t. In general $MDD \sim t^{\gamma}$ for some $\gamma > 0$. Codling et al, (143), discuss values of γ for several types of random walks. The standard relation between MSD and the time is $\gamma = 1$. However, other random walks such as Continuous Random Walks (CRW) have values of $1 < \gamma < 2$. These values of γ produce *super-diffusion* patterns and can be modeled using Lévy walk processes - step lengths have infinite variance. Kim et al, (144) analyze several human GPS mobility traces showing super-diffusive behavior and its impact on contact-based metrics.

Here, we still keep the Random Direction as mobility model in order to compare the FKPP PDE model with the previous ODE SI model. A key question is how to calculate the diffusion coefficient. For that, we have adopted the following approximation, based on simulation. Let us define an area of size A and N users moving with random direction in that area. We randomly choose one node to initiate the infection and we consider that $P_{c_2} = 1$ (i.e. nodes do not use power saving). Each time an infected

node meets another node - this node is in the infected node radio coverage area - it infects its (i.e., gives her a copy of the DTO with probability 1). We will obtain the diffusion coefficient D drawing,Figure 7.12, the *dissemination time* T_D as a function of the proportion of nodes, $s(t)$, that have a copy of the DTO at time t.

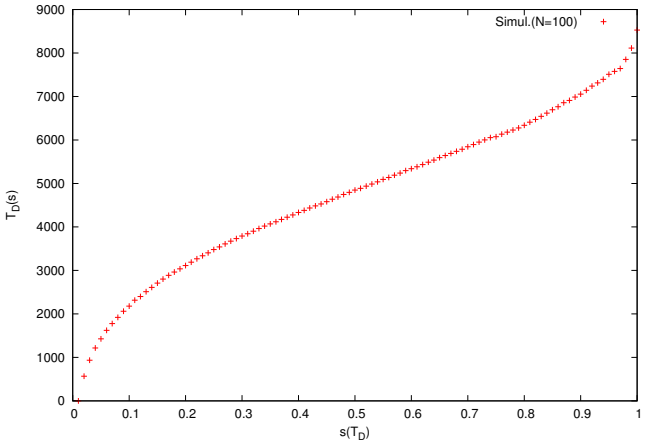


Figure 7.12: Dissemination time versus percentage of infected nodes, N=100. β=1.58x10^{-3}.

For that purpose, without loss of generality, we consider that the infection begins in a point of the area A and let us assume that a wave is propagated from this point. After a time T_D, the front of the propagating wave has occupied an area A_x. Let R_c be the radius of the propagating wave front and N_x the number of nodes inside area A_x. If the nodes are uniformly distributed in the area, the density λ_{N_x} of infected nodes in area A_x is equal to the density λ_N of nodes in the area A. Then, the proportion of infected nodes $\frac{N_x}{N}$ is equivalent to the proportion of area infected: $\frac{N_x}{N}$=$\frac{A_x}{A}$. Furthermore, $\frac{N_x}{N} \leq s(T_D)$. Then, the maximum radius of the propagating wave front $R_{c_{max}}$:

$$R_{c_{max}} = \{R_c : A_x = As(T_D)\} \tag{7.15}$$

For example, if the area is a square of width a, and the focus is initiated in a corner of the square: $A_x = \pi R_c^2/4$ if $R_c \leq a^2$ and $A_x = a^2 + (\frac{\pi}{4} - 1)((a - (R_c^2 - a^2)^{1/2})^2$ otherwise. Now, taking an operation pair point ($s(T_D)$,T_D) from Figure 7.12 we have to calculate which R_c fulfills equation 7.15. Once we have obtained the radius of the propagating wave front, $R_{c_{max}}$, we can calculate the wave speed as:

$$v_{min} = \frac{R_{c_{max}}|_{s(T_D)}}{T_D|_{s(T_D)}} \tag{7.16}$$

Now, since v_{min}=$\sqrt{\beta D}$=$\sqrt{2RV_eD\lambda_N}$, the diffusion coefficient D can be easily found.

The simulator (Figure 7.12) considers average values in which the infection begins in any part of the area. Then, we have also considered in the geometric model defined by equations (7.15) and (7.16) that the infection begins at different points of the area and we have averaged the value of the wave speed over all the values obtained. From this average, we have obtained D=29 that is quite approximate to the value D=28 obtained by Klein et al. in (142) using MonteCarlo simulations.

Once the diffusion coefficient D is known, the next step is to introduce the power saving constraints into this model. To do that, the following consideration enters into play: as in the ODE model, under the power saving mode of operation, nodes are part of the time with the wireless card off so the positive contact rate is no longer β but $\beta' = P_{c_2}\beta$. So, under the power saving constraints, v_{min}=$\sqrt{\beta' D}$. We validate this assumption through simulation. To validate this model we have performed simulations with the following conditions:

- Area, A = 4 Km^2
- Number of nodes, N = 100
- Period, T = 30 seconds
- Mobility model = Random Direction

Figure 7.13 presents the results for T_{srch} ranging from 30 seconds (i.e. nodes have the wireless card on 100% of the time) to 15 seconds (i.e. nodes have the wireless card on 50% of the time). As it can be seen, the dissemination time is not affected by T_{srch}. This is because in all cases P_{C_2}=1 (the analytical model is also plotted to show this point), meaning that the system is not affected by this parameter.

Recall from Table 7.2 that in order to have P_{C_2}=1 it is needed that $T_{sleep} \leq T_{srch}$, as we have for all cases, but also $2T_{sleep} \leq T_C$. To corroborate that the last condition holds, Figure 7.14 plots the CDF of the contact duration in the simulations. As it can be seen, only about 0.05% of the contacts take less than 30 seconds, making this second condition to be true for almost all the contacts in the four considered scenarios.

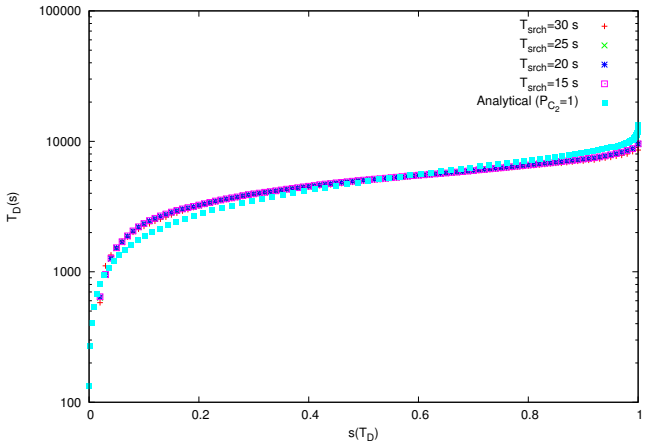


Figure 7.13: Dissemination time versus percentage of infected nodes, N=100. T=30.

Moreover, to really verify the model it is needed to show what happens when P_{C_2} <1. This can be done by making $T_{srch} \leq T_{sleep}$, thus moving the operation point from region E_2 to region D_2 in Figure 7.5. In this region D_2, $P_{C_2} = 2T_{srch}/T$. Figure 7.15 plots an example of such a scenario, in this case for T_{srch}=12 s. As it can be seen, the FKPP PDE model also fits the simulation results when considering contact probabilities lower than 1. This result extends the ones reported in (142) where no power saving constraints were considered.

It is needed to point out, however, that a further decrease in T_{srch} makes the FKPP PDE model to break down. This is because the model is based in a *traveling wave* that needs a minimum node density to propagate (for the area considered, this critical density is achieved for $N > 25$, see (142)).So, although the number of nodes (i.e. the density) is kept, if the power saving policy becomes very aggressive (i.e. nodes are with the wireless card off most of the time) the model fails to represent the system behaviour at some point. To identify this point, let us define the concept of *effective node density* as $\lambda'_N = \frac{N}{A} \cdot \frac{T_{srch}}{T}$ (note that the last fraction is the duty cycle). There is a value for λ'_N that marks the frontier between the regions where the network becomes well-mixed or not (i.e. the application of one or the other model is required).

For the considered scenario, this point is $\lambda'_N = \frac{25 nodes}{4Km^2}$. In other words, as N=100 is considered, a duty cycle lower than $\frac{1}{4}$ will imply that the FKPP PDE model is not valid and the model for the well-mixed case will apply better.

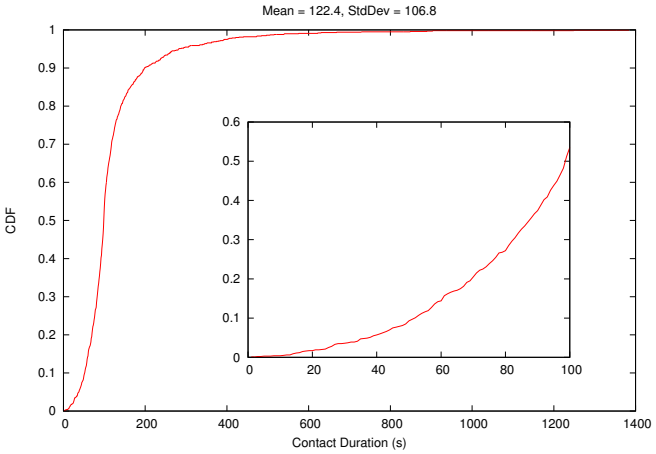


Figure 7.14: CDF of the contact duration

Figure 7.16 shows this behavior. Note that $T_{srch} = 5$ implies a duty cycle of $\frac{5}{30} = \frac{1}{6}$, or, in other words, a number of *effective nodes* equal to $\frac{100}{6} \backsimeq 17$ which indicates that the FKPP PDE model is not applicable. The subplot inside the figure illustrates how while λ'_N is kept above the frontier value, even very close to it, the FKPP PDE model matches the simulation results (note that the duty cycle is approximately 0.27, just a bit higher than the 0.25 mentioned before).

7.7 Conclusions

In this chapter, power saving trade-offs in DTNs as a function of the searching and sleeping intervals and as a function of node contact duration are investigated. Different operation regions are identified in which a node will have high contact probabilities with both other nodes and/or the infrastructure while allowing energy savings. Simulation results from pedestrian mobility traces show that the mathematical model is accurate and that the contact rates in these kind of networks can be quite low, thereby justifying the need for the nodes to switch off their wireless cards in accordance with the operation regions. We think that the modeling gives a hint on designing strategies to save batteries in this kind of scenarios.

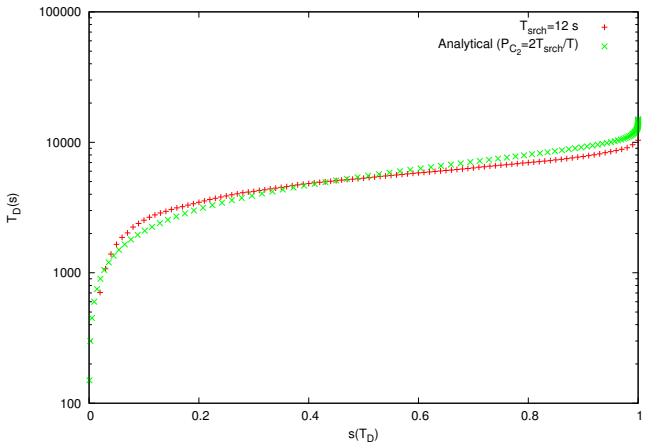


Figure 7.15: Dissemination time versus percentage of infected nodes, N=100. T=30.

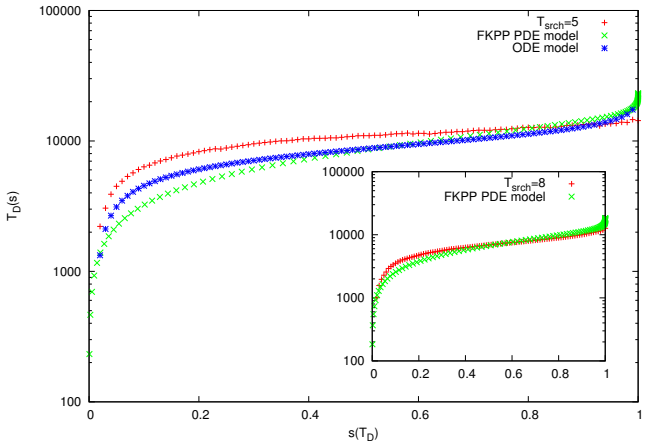


Figure 7.16: Dissemination time versus percentage of infected nodes, N=100. T=30.

8

Conclusions and Results

Information Dissemination in Mobile Networks is a novel and wide topic, with many open problems. Many issues remain to be discussed and solved before generic solutions can be deployed. This thesis has focused on some of those problems, defining some key points and ideas to face the Dissemination in Mobile Networks' open problems.

This is the summary of conclusions and results we have obtained in each of the topics we have dealt during this Thesis:

- **VANETS can benefit from vehicular cooperation** to improve the performance of the Mobile Ad hoc Network.

  Using transportation data and statistics, cooperation between nodes in an urban or suburban environment can be exploited to maximize the road infrastructure usage, as well as, to improve the vehicles' throughput.

  We analyzed various options with different degrees of synchronization among the infrastructure nodes, and different policies for the carrier, destination and data scheduling to be sent. Showing how each of these, affect Network performance parameters such as Cooperative Rate, AP load, overhead (data redundancy) or undelivery ratio. This characterization of the policies can help others to design their platforms and applications considerng the network performance parameters they expect from the network.

  In summary, the conclusion is that it is feasible to use simple mechanisms to reuse underutilized Road Infrastructure resources, as well as currently unexploited V2V opportunities.

- **Planning RSU deployments in city wide scenarios using statistical data of the Vehicular Mobility .** As one would expect, the more informed the algorithms used to plan RSU deployments the better tuning of the system can be achieved, while we have also proven how just some aggregated information or statistics about Vehicular Mobility in cities and urban or interurban scenarios can help to achieve RSU deployments that would disseminate effectively data contents to most of the vehicles as fast as possible.

  We have shown how this data can be used by algorithms that are NP-Hard to obtain the optimal deployment to maximize Data dissemination, and how, using heuristics we can achieve very similar solutions.

  In summary, the conclusion is that using Transportation authorities' information about common vehicular trajectories, it is possible to disseminate data to a high ratio of vehicles with a few dissemination points placed according to our deployment heuristics.

- **Malware in VANETS can spread rapidly, and pose a real threat**, while it's not possible to predict the propagation features and characteristics of such a malware, we have proposed a model to simulate or predict its spreading behaviour for a wide range of parameters.

  We have shown that it is possible to obtain a model that uses per road statistics, already gathered by transportation authorities, to describe the malware spread in urban/suburban scenarios. Validating it against simulations that make use of detailed vehicular Mobility traces. Concluding that using the model can help others to analyze the malware spreading in cities with no mobility traces available, and to do it much faster with the model than running the mobility trace simulations.

- **Trade-off between power saving techniques in MANET nodes and the contact opportunities missed**. We have studied the trade-off between the On/Off duty-cycle techniques (used to save power during the Off periods), and the missed contact opportunities (that are so important in MANETS).

  We have defined the analytical model for the contact probability. Both for the case where both nodes are using On/Off duty-cycle techniques, and for the case where one of the nodes is always On.

In summary, the conclusion is that it is possible to use the model to limit the missed contact opportunities' probability that the designer of an application is willing to accept in compensation for an increased battery life due to the On/Off duty-cycles.

- **Network Coding decoding probability error** Network Coding is used in many frameworks and projects, and there are many issues to consider when designing a solution using it. One of them is the decoding probability error. This error is small when using NC under large Galois Fields. Many works so far, have neglected it or used an approximated upper bound error to build their frameworks, because there were no better (more precise) solution.

  We have calculated this decoding error probability, defining the exact formulation to find this value. This allows to define the number of extra packets that must be sent to avoid this decoding error, as a function of the number of packets and the Galois field used. This result has already proven to be useful for other people using Network Coding.

Bibliography

[1] Cisco. **Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2012–2017**. In *White paper*, February 2013. 2, 3

[2] T. Clausen and P. Jacquet. **RFC 3626: Optimized Link State Routing Protocol (OLSR)**, Oct 2003. 4

[3] C. Perkins, E. Belding-Royer, and S. Daas. **RFC 3561: Ad hoc On-Demand Distance Vector Routing**, Jul 2003. 4

[4] D. B. Johnson, D. A. Maltz, and Y. Hu. **DSR-draft: The Dynamic Source Routing Protocol for Mobile Ad Hoc Netowrks (DSR)**, April 2003. 4

[5] M. Abolhasan, T. Wysocki, and E. Dutkiewicz. **A review of routing protocols for mobile ad hoc networks**. In *Journal of Ad Hoc Networks, Vol 2, pp 1-22*, 2004. 4

[6] J. Ott and D. Kutscher. **Drive-thru Internet: IEEE 802.11b for 'Automobile' users**. In *IEEE 23th Conference on Computer Communications (INFOCOM), Hong Kong*, March, 2004. 5, 32

[7] X. Zhang, J. Kurose, B. Levine, D. Towsley, and H. Zhang. **Modeling of a Bus-based Disruption Tolerant Network Trace**. In *The Thirteenth Annual International Conference on Mobile Computing and Networking (MOBICOM), Montreal, Canada*, Sept, 2007. 5

[8] J. Burguess, B. Gallagher, D. Jensen, and B.N. Levine. **MaxProp: Routing for Vehicle-based Disruption Tolerant Networks**. In *25th Conference on Computer Communications (IEEE INFOCOM'05), Barcelona, Spain*, May 2006. 5, 30

[9] K. Fall. **A delay-tolerant network architecture for challenged internets**. In *ACM SIGCOMM 2003*, Aug. 2003. 7, 8, 29

[10] L. M. Feeney and M. Nilsson. **Investigating the energy consumption of a wireless network interface in an ad hoc networks** . In *IEEE Conf. on Computer Communications (Infocom'01)*, April 2001. 8, 140

[11] R. Ahlswede, N. Cai, S.Y.R. Li, and R.W. Yeung. **Network Information Flow**. In *IEEE Transactions on Information Theory, vol. 46, no. 4, pp. 1204-1216*, 2000. 8, 69

[12] T. Ho, M. Medard, R. Koetter, D.R. Karger, M. Effros, J. Shi, and B. Leong. **A Random Linear Network Coding Approach to Multicast**. In *IEEE Transactions on Information Theory, vol. 52, no. 10, pp. 4413-4430*, 2006. 8, 69, 71

[13] T. Ho, B. Leong, M. Medard, R. Koetter, Y. Chang, and M. Effros. **On the utility of network coding in dynamic environments**. In *IWWAN'04, Oulu, Finland*, 2004. 8, 69

[14] J. Widmer and J.-Y. Le Boudec. **Network coding for efficient communication in extreme networks**. In *ACM WDTN'05, Philadelphia, USA*, 2005. 8, 69

[15] D. Lucani, M. Medard, and M. Stojanovic. **Random Linear Network Coding for Time Division Duplexing: When to Stop Talking and Start Listening** . In *IEEE INFOCOM'09, Rio de Janeiro, Brazil*, 2009. 8, 69

[16] L. Rass and J. Radcliffe. **Spatial Deterministic Epidemics**. In *Mathematical Surveys and Monographs, Vol 102, American Mathematical Society*, 2003. 9

[17] R. Groenevelt, P. Nain, and G. Koole. **The message delay in mobile ad hoc networks**. In *Performance Evaluation (Elsevier), Vol 62, issue 1-4, pp. 210–228*, October 2005. 9

[18] X. Zhang, G. Neglia, J. Kurose, and D. Towsley. **Performance modeling of epidemic routing**. In *Computer Networks, 51, issue 10,(2007), 2867-2891*, July 2007. 9, 156, 157

[19] T Nadeem, P. Shankar, and L. Iftode. **A Comparative Study of Data Dissemination Models for VANETs**. In *3rd Annual International Conference on Mobile and Ubiquitous Systems (MOBIQUITOUS), San Jose, California*, July 2006. 9

[20] I. Leontiadis and C. Mascolo. **Opportunistic Spatio-Temporal Dissemination System for Vehicular Networks**. In *ACM MobiOpp, San Juan, Puerto Rico, USA*, June 2007. 9, 112

[21] C. Lochert, B. Scheuermann, M. Caliskan, and M. Mauve. **The Feasibility of Information Dissemination in Vehicular Ad-Hoc Networks**. In *WONS, Obergurgl, Austria*, January 2007. 9, 112

[22] M. Nekovee. **Epidemic algorithms for reliable and efficient information**. In *dissemination in vehicular ad hoc networks*, IET Intelligent Transport Systems, 3(2):104–110. 9

[23] S. Panichpapiboon and W. Pattara-atikom. **A Review of Information Dissemination Protocols for Vehicular Ad Hoc Networks**. In *IEEE Comm. Surveys and Tutorials*, 2011. 9

[24] Y.P. Fallah, C.-L. Huang, R. Sengupta, and H. Krishnan. **Analysis of Information Dissemination in Vehicular Ad-Hoc Networks with Application to Cooperative Vehicle Safety Systems**. In *IEEE Trans. Vehicular Technology, 60(1):232–247*, January 2011. 9, 111

[25] P. Ferrie, P. Szor, R. Stanev, and R. Mouritzen. **Security responses: Symbos.cabir**. In *Symantec Corporation*, 2004. 10, 107

[26] E. Chien. **Security response: Symbos.mabir**. In *Symantec Corporation*, 2005. 10, 108

[27] M. Lactaotao. **Security information: Virus encyclopedia: Symbos comwar.a: Technical details**. In *Trend Micro Incorporated*, 2005. 10, 108

[28] J. Kleinberg. **The wireless epidemic**. In *Nature, 449:287-288*, Sept 2007. 10, 108

[29] E. Anderson, K. Eustice, S. Markstrum, M. Hanson, and P. Reiher. **Mobile Contagion: Simulation of Infection & Defense**. In *Symp. on Measurement, Modelling, and Simulation of Malware, Monterey, USA*, 2005. 10, 108

[30] J. Su, K.W. Chan, A.G. Miklas, K. Po, A. Akhavan, S. Saroiu, E. de Lara, and A. Goel. **A Preliminary Investigation of Worm Infections in a Bluetooth Environment**. In *ACM WORM, Fairfax, VA, USA*, 2006. 10, 108

[31] P. Akritidis, C.W. Yung, V.T. Lam, S. Sidiroglou, and K.G. Anagnostakis. **Proximity Breeds Danger: Emerging Threats in Metro-area Wireless Networks**. In *USENIX Security, Boston, MA, USA*, 2007. 10, 108

[32] P. Wang, M.C. Gonzalez, C.A. Hidalgo, and A.-L.Barabasi. **Understanding the spreading patterns of mobile phones viruses**. In *Science 324:1071–1076*, May 2009. 10, 108

[33] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, and S. Savage. **Experimental Security Analysis of a Modern Automobile**. In *IEEE S&P, Oakland, CA*, 2010. 10, 108

[34] I. Rouf, R. Miller, H. Mustafa, T. Taylor, S. Oh, W. Xu, M. Gruteser, W. Trappe, and I. Seskar. **Security and Privacy Vulnerabilities of In-Car Wireless Networks: A Tire Pressure Monitoring System Case Study**. In *USENIX Security, Washington, DC, USA*, 2010. 11, 108

[35] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, and T. Kohno. **Comprehensive Experimental Analyses of Automotive Attack Surfaces**. In *USENIX Security*, 2011. 11, 108

[36] S. Uppoor, O. Trullols-Cruces, M. Fiore, and J.M. Barcelo-Ordinas. **Generation and Analysis of a Large-scale Urban Vehicular Mobility Dataset**. In *IEEE Transactions on Mobile Computing, published online, to be printed*, 2013. viii, 11, 14, 26

[37] H. Meyer, **O. Trullols**, A. Hess, K.A. Hummel, J.M. Barcelo, C.E. Casetti, and G. Karlsson. **VANET Mobility Modeling Challenged by Feedback Loops (Invited Paper)**. In *10th IEEE/IFIP Annual Mediterranean Ad Hoc Networking Workshop (IEEE/IFIP Med-Hoc-Net 2011)*, 2011. viii, 11, 26

[38] Marco Gramaglia, Oscar Trullols-Cruces, Diala Naboulsi, Marco Fiore, and Maria Calderon. **Vehicular Networks on Two Madrid Highways**. In *Under Review*, 2014. 11

[39] T. Camp, J. Boleng, and V. Davies. **A survey of mobility models for ad hoc network research**. In *Wireless Communication and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications, vol. 2, no. 5, pp. 483–502*, 2002. 12

[40] Kyunghan Lee, Seongik Hong, Seong Joon Kim, Injong Rhee, and Song Chong. **SLAW: A New Mobility Model for Human Walks**. In *INFOCOM 2009, IEEE*, 2009. 13

[41] S.P. Hoogendoorn and P.H.L. Bovy. **Pedestrian route-choice and activity scheduling theory and models**. In *Elsevier Transportation Research Part B*, 2003. 13

[42] J. Kim, V. Sridhara, and S. Bohacek. **Realistic mobility simulation of urban mesh networks** . In *Ad Hoc Networks, 7, issue 2, Pages 411-430*, March 2009. 13, 153

[43] **UDel Models For Simulation of Urban Mobile Wireless Networks**. 13

[44] J. Harri, F. Filali, and C. Bonnet. **Mobility Models for Vehicular Ad Hoc Networks: A Survey and Taxonomy**. In *Eurecom Sophia-Antipolis, RR-06-168*, March 2007. 13

[45] M. Fiore and J. Härri. **The networking shape of vehicular mobility**. In *ACM MobiHoc'08, Hong Kong, China*, May 2008. 13, 59

[46] J. Morillo. **Improving Wireless Networks with Cooperative ARQ Protocols**. In *Thesis, Departament d'Arquitectura de Computadors*, Barcelona, Spain, 14 July 2009. 14

[47] J. Morillo, J. M. Barcelo, O. Trullols, and J. Garcia-Vidal. **A Cooperative ARQ for Delay-Tolerant Vehicular Networks**. In *published in DTMN 2008*, 2008. 14, 23

[48] J. Morillo, **O. Trullols**, J. M. Barcelo, and J. Garcia-Vidal. **Evaluation of a Cooperative ARQ Protocol for Delay-Tolerant Vehicular Networks**. In *Wireless and Mobility 2008, LNCS 5122 proceedings*, 2008. 14, 23

[49] J. Morillo, **O. Trullols**, J. M. Barcelo, and J. Garcia-Vidal. **Applying Cooperation for Delay Tolerant Vehicular Networks**. In *Fourth EuroFGI Workshop on Wireless and Mobility 2008*, Barcelona, Spain January 2008. 14, 23

[50] **O. Trullols**, J. Morillo, J. M. Barcelo, and J. Garcia-Vidal. **A Cooperative Vehicular Network Framework**. In *ICC '09 IEEE International Conference on Communications, Dresden, Germany*, 2009. vii, 15, 22, 29

[51] **O. Trullols**, M. Fiore, and J.M. Barcelo. **Cooperative download in vehicular environments**. In *IEEE Transactions on Mobile Computing (TMC)*, 2011. vii, 15, 18, 22

[52] **O. Trullols**, M. Fiore, C. Casetti, C.F. Chiasserini, and J.M. Barcelo. **Information Dissemination in VANETs: Deployment Strategies for Maximizing Coverage**. In *Euro-NF Workshop on Wireless and Mobility, Stresa, Lake Maggiore, Italy* , 2009. 15

[53] **O. Trullols**, M. Fiore, C. Casetti, C.F. Chiasserini, and J.M. Barcelo. **A Max Coverage Formulation for Information Dissemination in Vehicular Networks**. In *IEEE Int. Conf. on Wireless and Mobile Computing, Networking and Communications,Marrakech, Morocco*, 2009. viii, 15, 24

[54] **O. Trullols**, M. Fiore, C. Casetti, C.F. Chiasserini, and J.M. Barcelo. **Planning Roadside Infrastructure for Information Dissemination in Intelligent Transportation Systems**. In *Computer Communications, Elsevier*, 2010. vii, 15, 18, 24, 30

[55] M. Reineri, C. Casetti, C.F. Chiasserini, M. Fiore, **O. Trullols**, and J.M. Barcelo. **RSU Deployment for Content Dissemination and Downloading in Intelligent Transportation Systems**. In *Book chapter "Roadside Networks for Vehicular Communications: Architectures, Applications and Test Fields"*, 2012. viii, 15, 24

[56] O. Trullols-Cruces, M. Fiore, and J.M. Barcelo-Ordinas. **Understanding, modeling and taming mobile malware epidemics in a large-scale vehicular network**. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium*, 2013. viii, 15, 18, 25

[57] **O. Trullols**, J. Morillo, J. M. Barcelo, and J. Garcia-Vidal. **Power Saving Trade-offs in Delay/Disruptive Tolerant Networks**. In *IEEE WoWMoM 2011, Lucca*, 2011. viii, 15, 18, 25

[58] **O. Trullols**, J. Morillo, and J. M. Barcelo. **Impact of the Infrastructure in Mobile Opportunistic Networks(Invited Paper)**. In *4th International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL 2011)*, 2011. 15

[59] **O. Trullols**, J.M. Barcelo, and M. Fiore. **Exact Decoding Probability Under Random Linear Network Coding**. In *IEEE Communication Letters, January 2011, ISSN:1089-7798*, 2010. viii, 15, 18, 26

[60] Marco Fiore and Jose M. Barcelo-Ordinas. **Cooperative Download in Urban Vehicular Networks**. In *6th IEEE International Conference on Mobile Ad Hoc and Sensor Systems, (IEEE MASS 2009), Macau, China*, October 2009. 22

[61] F. Aidouni, M. Latapy, and C. Magnien. **Ten weeks in the life of an eDonkey server**. In *Hot-P2P'09, Rome, Italy,* , May 2009. 28

[62] A. Nandan, S. Das, G. Pau, M. Gerla, and M.Y. Sanadini. **Co-operative Download in Vehicular Ad Hoc Wireless Networks**. In *2nd annual conference on Wireless On-demand Network Systems and Services, WONS'05,Italy*, 2005. 29

[63] M. Sardari, F. Hendessi, and F. Fekri. **Infocast: A New Paradigm for Collaborative Content Distribution from Roadside Units to Vehicular Networks**. In *IEEE SECON'09, Rome, Italy*, June 2009. 29

[64] B.B. Chen and M.C. Chan. **MobTorrent: A Framework for Mobile Internet Access from Vehicles**. In *IEEE INFOCOM'09, Rio de Janeiro, Brasil*, April 2009. 30

[65] J. Zhao and G. Cao. **VADD: Vehicle-Assisted Data Delivery in Vehicular Ad Hoc Networks**. In *25th Conference on Computer Communications (IEEE INFOCOM'06), Barcelona, Spain*, April, 2006. 30

[66] S. Yoon, H.Q. Ngo, and C. Qiao. **On 'Shooting' a Moving Vehicle with Data Flows**. In *IEEE MOVE'07, Alaska, USA*, May 2007. 30

[67] F. Malandrino, C. Casetti, C.-F. Chiasserini, and M. Fiore. **Content Downloading in Vehicular Networks: What Really Matters**. In *IEEE INFOCOM'11, Shangai, China*, April 2011. 30

[68] Z. Chen, H. Kung, and D. Vlah. **Ad hoc relay wireless networks over moving vehicles on highways**. In *ACM MobiHoc'01, Long Beach, USA*, October 2001. 30

[69] H.-Y. Huang, P.-E. Luo, M. Li, D. Li, X. Li, W. Shu, and M.-Y. Wu. **Performance evaluation of SUVnet with real-time traffic data**. In *IEEE Transactions on Vehicular Technology, vol. 56, no. 6*, November 2007. 30

[70] H. Wu, R. Fujimoto, R. Guensler, and M. Hunter. **MDDV: a mobility-centric data dissemination algorithm for vehicular networks**. In *ACM VANET'04, Philadelphia, USA*, October 2004. 30

[71] A. Skordylis and N. Trigoni. **Delay-bounded routing in vehicular ad hoc networks**. In *ACM MobiHoc'08, Hong Kong, China*, May 2008. 30

[72] J. Zhang, Q. Zhang, and W. Jia. **A novel MAC protocol for cooperative downloading in vehicular networks**. In *IEEE GLOBECOM'07, Washington, USA*, December 2007. 30

[73] S. Ahmed and S.S. Kanhere. **VANETCODE: network coding to enhance cooperative downloading in vehicular ad hoc networks**. In *ACM IWCMC'06, Vancouver, Canada* , July 2006. 30

[74] G. Marfia, G. Pau, E. Giordano, E. De Sena, and M. Gerla. **Evaluating vehicle network strategies for downtown Portland: opportunistic infrastructure and importance of realistic mobility models**. In *ACM MoBiOpp'07, San Juan, Puerto Rico*, June 2007. 30

[75] Y. Ding, C. Wang, and L. Xiao. **A Static-Node Assisted Adaptive Routing Protocol in Vehicular Networks**. In *ACM VANET'07, Montreal, Canada*, September 2007. 30

[76] C. Lochert, B. Scheuermann, C. Wewetzer, A. Luebke, and M. Mauve. **Data aggregation and roadside unit placement for a vanet traffic information system**. In *ACM VANET'08, S.Francisco, USA*, September 2008. 30, 81

[77] Z. Zheng, P. Sinha, and S. Kumar. **Alpha Coverage: Bounding the Interconnection Gap for Vehicular Internet Access**. In *IEEE INFOCOM 2009, Rio de Janeiro, Brasil*, April 2009. 30

[78] Z. Zheng, Z. Lu, P. Sinha, and S. Kumar. **Maximizing the Contact Opportunity for Vehicular Internet Access**. In *IEEE INFOCOM 2010, San Diego, USA*, March 2010. 30

[79] A. Fleisher. **On prediction and urban traffic**. In *Papers in Regional Science, vol.7, no. 1*, December 1961. 32, 48

[80] K. Ashok and M.E. Ben-Akiva. **Estimation and prediction of time-dependent origin-destination flows with a stochastic mapping of path flows and link flows**. In *Transportation Science, vol. 36, no. 2*, May 2002. 32, 48

[81] H. Yin, S.C. Wong, J. Xu, and C.K. Wong. **Urban traffic flow prediction using a fuzzy-neural approach**. In *Transportation Research C, vol. 10, no. 2*, April 2002. 32, 48

[82] ETH Zurich. **ETH Zurich Realistic Vehicular Traces**. 46, 80, 94

[83] N. Cetin, A. Burri, and K. Nagel. **A large-scale multi-agent traffic microsimulation based on queue model**. In *STRC'03, Ascona, Switzerland*, March 2003. 46, 94

[84] J. Bicket, D. Aguayo, S. Biswas, and R. Morris. **Architecture and Evaluation of an Unplanned 802.11b Mesh Network**. In *ACM MobiCom'05, Koln, Germany*, August 2005. 47

[85] V. Bychkovsky, B. Hull, A. K. Miu, Hari Balakrishnan, and Samuel Madden. **A measurement study of vehicular internet access using in situ wi-fi networks**. In *12th annual international conference on Mobile computing and networking (MobiCom '06)*, 2006. 47

[86] **O. Trullols**. **VANET chunk scheduling simulator**. 52

[87] R. Gass, J. Scott, and C. Diot. **Measurements of in-motion 802.11 networking**. In *IEEE WMCSA/HotMobile'06, Washington, USA*, April 2006. 53

[88] F. Hui and P. Mohapatra. **Experimental Characterization of Multi-hop Communications in Vehicular Ad Hoc Networks**. In *ACM VANET'05, Koln, Germany*, September 2005. 54

[89] D. Lucani, M. Medard, and M. Stojanovic. **Random Linear Network Coding for Time Division Duplexing: Field Size Considerations** . In *IEEE GLOBECOM'09, Hawaii, USA*, 2009. 70, 75, 76

[90] E. Balas. **A class of Location, Distribution and Scheduling Problems: Modeling and Solution Methods**. In *Chinese-US Symposium on System Analysis, Wiley and Sons*, 1983. 81

[91] S. Ceria, P. Nobili, and A. Sassano. **Set Covering Problem: Annotated bibliographies in Combinatorial Optimization**. In *Wiley and Sons, New York, pp., 415-428*, 1998. 81

[92] V.T. Paschos. **A survey of Approximately Optimal Solutions to some Covering and Packing Problems**. In *ACM Computing Surveys, vol., 29, no., 2, pp., 171-209*, 1997. 81

[93] T. Grossman and A. Wool. **Computational experience with approximation algorithms for the set covering problem**. In *European Journal of Operational Research, vol., 101, pp., 81-92*, 1997. 81

[94] A. Caprara, M. Fischetti, and P. Toth. **Algorithms for the Set Covering Problem**. In *Technical Report No.OR-98-3, DEIS - Operations Research Group*, 1998. 81

[95] D. Pisinger. **Where are the Hard Knapsack Problems?** In *Computers and Operations Research, vol., 32, pp., 2271-2284*, 2004. 81, 89

[96] M.H. Wright. **Optimization methods for base station placement in wireless applications**. In *IEEE VTC, Ottawa, Canada*, 1998. 81

[97] R. M. Whitaker, L. Raisanen, and S. Hurley. **The Infrastructure Efficiency of Cellular Wireless Networks**. In *Computer Networks, vol., 48, no., 6, pp., 941–959*, 2005. 81, 83

[98] K. Tutschku. **Demand-based radio network planning of cellular mobile communication systems**. In *IEEE INFOCOM, San Francisco, USA*, 1998. 81

[99] E. Amaldi, A. Capone, M. Cesana, and F. Malucelli. **Optimizing WLAN radio coverage**. In *IEEE ICC, Madison, USA*, 2004. 81

[100] A. Chaintreau, J.Y. Le Boudec, and N. Ristanovic. **The age of gossip: Spatial mean-field regime**. In *ACM SIGMETRICS 2009, Seattle, WA*, 2009. 82

[101] R.H. Frenkiel, B.R. Badrinath, J. Borras, and R. Yates. **The Infostations Challenge: Balancing Cost and Ubiquity in Delivering Wireless Data**. In *IEEE Personal Communications Magazine, 7(2): pp. 66-71,*, April 2000. 82

[102] G. Sollazo, M. Musolesi, and C. Mascolo. **TACO-DTN: a Time-Aware COntent-based dissemination system for Delay Tolerant Networks**. In *ACM MobiOpp, Puerto Rico, USA*, 2007. 82

[103] R Cohen, D. Raz, and M. Aezladen. **Locally vs. Globally Optimized Flow-Based Content Distribution to Mobile Nodes**. In *IEEE INFOCOM, Rio de Janeiro, Brazil*, 2009. 82

[104] H. Kim, Y. Seok, N. Choi, Y. Choi, and T. Kwon. **Optimal Multi-sink Positioning and Energy-efficient Routing in Wireless Sensor Networks**. In *International Conference on Information Networking (ICOIN)*, 2005. 82, 83

[105] E. I. Oyman and C.Ersoy. **Multiple Sink Network Design Problem in Large Scale Wireless Networks**. In *IEEE ICC*, June 2004. 82, 83

[106] W. Y. Poe and J. B. Schmitt. **Minimizing the Maximum Delay in Wireless Sensor Networks by Intelligent Sink Placement**. In *Technical Report 362/07 University of Kaiserslautern, Germany*, July 2007. 82, 83

[107] Y. Hu, Y. Xue, Q. Li, F. Liu, G. Y. Keung, and B. Li. **The Sink Node Placement and Performance Implication in Mobile Sensor Networks**. In *Journal on Mobile Networks and Applications (MONET) vol 14, pp 230–240*, Feb 2009. 82, 83

[108] K. Maeda, A. Uchiyama, T. Umedu, H. Yamaguchi, K. Yasumoto, and T. Higashino. **Urban pedestrian mobility for mobile wireless network simulation**. In *Ad Hoc Networks, vol., 7, pp., 153-170*, 2009. 82

[109] S. O. Krumke, M. V. Marathe, D. Poensgen, S. S. Ravi, and H.-C. Wirth. **Budgeted Maximum Graph Coverage**. In *Lecture Notes In Computer Science, vol., 2573, pp., 321–332*, 2002. 83

[110] L. Qiu, R. Chandra, K. Jain, and M. Mahdian. **Optimizing the Placement of Integration Points in Multi-Hop Wireless Networks**. In *International Conference on Network Protocols (ICNP) Berlin, Germany*, Oct 2004. 83

[111] **Eurecom VanetMobiSim**. 84

[112] M. Fiore, J. Harri, F. Fethi, and C. Bonnet. **Vehicular mobility simulation for VANETs**. In *Proc. of the 40th IEEE Annual Simulation Symposium (ANSS'07)*, March 2007. 84

[113] A. A. Ageev and M. I. Sviridenko. **Approximation Algorithms for Maximum Coverage and Max Cut with Given Sizes of Parts**. In *Lecture Notes in Computer Science, vol, 16*, 1999. 87

[114] J.W. Mickens and B.D. Noble. **Modeling Epidemic Spreading in Mobile Environments**. In *ACM WiSe, Cologne, Germany*, 2005. 110

[115] S.A. Khayam and H. Radha. **Analyzing the Spread of Active Worms over VANET**. In *ACM VANET, Philadelphia, PA*, 2004. 110, 134

[116] M. Nekovee. **Modeling the Spread of Worm Epidemics in Vehicular Ad Hoc-Networks**. In *IEEE VTC-Spring, Melbourne*, 2006. 110, 134

[117] L. Cheng and R. Shakya. **VANET Worm Spreading from Traffic Modeling**. In *IEEE RWS, New Orleans, LA, USA*, 2010. 110, 111

[118] M. Gramaglia, P. Serrano, J.A. Hernandez, M. Calderon, and C.J. Bernardos. **New Insights from the Analysis of Free Flow Vehicular Traffic in Highways**. In *IEEE WoWMoM, Lucca, Italy*, 2011. 110, 111, 124, 125

[119] L. Cheng and S. Panichpapiboon. **Effects of intervehicle spacing distributions on connectivity of VANET: a case study from measured highway traffic**. In *IEEE Communications Magazine, 50(10):90–97*, October 2012. 110, 111

[120] B. Hoh and M. Gruteser. **Computer Ecology: Responding to Mobile Worms with Location-Based Quarantine Boundaries**. In *WSPWN, Miami, FL, USA*, 2006. 111

[121] A. Agarwal, D. Starobinski, and T.D.C. Little. **Analytical Model for Message Propagation in Delay Tolerant Vehicular Ad Hoc Networks**. In *IEEE VTC-Spring, Singapore*, 2008. 111

[122] E. Baccelli, P. Jacquet, B. Mans, and G. Rodolakis. **Information propagation speed in bidirectional vehicular delay tolerant networks**. In *IEEE Infocom, Shanghai, China*, 2011. 111

[123] H. Wu, R.M. Fujimoto, G.F. Riley, and M. Hunter. **Spatial Propagation of Information in Vehicular Networks**. In *IEEE Trans. Vehicular Technology, 58(1):420–431*, January 2009. 111

[124] Z. Zhang, G. Mao, and B.D.O. Anderson. **On the Information Propagation Process in Mobile Vehicular Ad Hoc Networks**. In *IEEE Trans. Vehicular Technology, 60(5):2314–2325*, June 2011. 111

[125] M. Fiore. **Vehicular Mobility Models**. In *Vehicular Networks: from Theory to Practice Chapman and Hall/CRC*, 2009. 111

[126] N. Weaver, V. Paxson, S. Staniford, and R. Cunningham. **A Taxonomy of Computer Worms**. In *ACM WORM, Washington, DC, USA*, 2003. 112

[127] B. Raney, N. Cetin, A. Völlmy, M. Vrtic, K. Axhausen, and K. Nagel. **An agent-based microsimulation model of Swiss travel: First results**. In *Networks and Spatial Economics, 3(1):23–41*, January 2003. 116

[128] TomTom. **How TomTom's HD TrafficTM and IQ RoutesTM data provides the very best routing**. In *White paper*, 2010. 125

[129] N. Banerjee, M. D. Corner, and B. N. Levine. **An Energy-Efficient Architecture for DTN Throwboxes** . In *IEEE Conf. on Computer Communications (Infocom'07)*, May 2007. 141

[130] N. Mishra, K. Chebrolu, B. Raman, , and A. Pathak. **Wake-on-WLAN**. In *In Proc.Intl Conf on the World Wide Web (WWW), pages 761-769*, 2006. 141

[131] T. Pering, Y. Agarwal, R. Gupta, , and R. Want. **CoolSpots: Reducing the Power Consumption of Wireless Mobile Devices with Multiple Radio Interfaces**. In *In Proc. ACM MobiSys, pages 220-232*, June 2006. 141

[132] T. Pering, V. Raghunathan, , and R. Want. **Exploiting Radio Hierarchies for Power-Efficient Wireless Device Discovery and Connection Setup**. In *In VLSI Design, pages 774-779*, 2005. 141

[133] H. Hun, M. H. Ammar, M. D. Corner, and E. W. Zegura. **Hierarchical Power Management in Disruption Tolerant Networks with Traffic Aware Optimization**. In *In Proceedings of ACM SIGCOMM Workshop on Challenged Networks (CHANTS'06)*, Sept 2006. 141

[134] N. Banerjee, M. D. Corner, , and B. N. Levine. **An energy efficient architecture for throwboxes in disruption tolerant networks** . In *Technical Report 06-39, UMass Amherst*, 2006. 141

[135] N. Banerjee, M. D. Corner, and B. N. Levine. **Design and Field Experimentation of an Energy-Efficient Architecture for DTN Throwboxes**. In *IEEE/ACM Transactions on Networking, 18(2):554–567*, April 2010. 141

[136] W. Wang, V. Srinivasan, and M. Motani. **Adaptive Contact Probing Mechanisms for Delay Tolerant Applications** . In *13th annual ACM international conference on Mobile computing and networking*, (MobiCom'07), Montreal, Quebec, Canada. 142

[137] LAN MAN Standards Committee. **IEEE 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY)**. In *Specifications ANSI/IEEE Std 802.11*, 1999. 143

[138] E. Shih, P. Bahl, and M. Sinclair. **Wake on Wireless: An event-driven energy saving strategy for battery-operated devices**. In *In Proceedings of the 8th Int. Conf. on Mobile Computing and Networking (MobiCom'02), pp. 160–171*, Sept. 2002. 143

[139] A. Vahdat and D. Becker. **Epidemic Routing for Partially-Connected Ad Hoc Networks**. In *Technical Report CS 200006, Duke University*, Ap. 2000. 143

[140] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. **Impact of Human Mobility on the Design of Opportunistic Forwarding Algorithms**. In *IEEE Transactions on Mobile Computing, 6(6):606–620*, June 2007. 152

[141] N. Eagle and A. Pentland. **Reality mining: Sensing complex social systems**. In *Springer, Journal of Personal and Ubiquitous Computing, 10(4)*, May 2006. 152

[142] D. J. Klein, J. Hespanha, and U. Madhow. **A Reaction-Diffusion Model for Epidemic Routing in Sparsely Connected MANETs** . In *IEEE Conf. on Computer Communications (Infocom'10), San Diego, CA, USA*, March 2010. 156, 158, 160, 163, 164

[143] E. A. Codling, M. J. Plank, and S. Benhamou. **Random walk models in biology** . In *Journal of the Royal Society Interface, 5: 813-834*, 2008. 161

[144] S. Kim, C. Lee, and D. Young Eun. **Super-Diffusive Behavior of Mobile Nodes and its impact on Routing Protocol Performance** . In *IEEE Transactions on Mobile Computing, Vol 9, n 2, pp 288-304*, Feb 2010. 161