Quantification in Device-to-Device Networks : from Link Estimation to Graph Utility

Clément Bertier

Quantification in Device-to-Device Networks: From Link Estimation to Graph Utility

By
Clément Bertier

A Dissertation submitted for the degree of
DOCTEUR en INFORMATIQUE et TÉLÉCOMMUNICATIONS
of
Sorbonne Université

Committee:

Reviewers:	Katia Jaffres-Runser	Associate Professor, INPT-ENSEEIHT
	Fabrice Valois	Professor, INSA Lyon
Examiners:	Aline Carneiro Viana	Researcher, INRIA Saclay
	Pierre Sens	Professor, Sorbonne Université
Advisors:	Farid Benbadis	Researcher, Thales SIX GTS
	Vania Conan	Researcher, Thales SIX GTS
	Marcelo Dias de Amorim	Research Director, CNRS/Sorbonne Université

January 2020

LIP6 NPA

Abstract

Device-to-device (D2D) communications are valuable in several domains, such as data offloading and diffusion, as their cost is only a fraction of what regular cellular communication would have. With the ubiquity of smartphones, opportunities for direct communications between users are more likely than ever. In this thesis, we argue that *understanding the potential utility* behind direct communications is key to quantifying the realization of contact networks. We tackle related questions from two distinct, yet complementary contributions.

Firstly, we consider the problem of estimating the importance of a node in large dynamic topologies. We focus on the traditional node utility calculation in graphs, known as the centrality, applied in a contact network context. Computing any node's centrality in a dynamic context is challenging because of the perpetual creation and destruction of links between users. We propose a novel approach to estimate centralities based on a pre-established database, where the estimation is based on the geographical coordinates of the node instead of the identifier of the node. As the centrality of a coordinate is much more stable than that of a mobile node, we can achieve a good centrality estimation at a fraction of the cost of traditional strategies. Our results using real-world mobility datasets show that we can obtain an accurate estimation of the centrality of up to 80% of the nodes with our method.

Estimating the centrality of a node alone is not enough to characterize the node's communication capabilities. In our second contribution, we propose to quantify the value of direct links through an experimental measurement campaign. We conceived a tool to actively measure the throughput between different Android devices, using available device-to-device APIs. We derived a model to obtain an estimate of the upper-bound of D2D throughput based on the distance between the devices. Besides the model, we could also leverage a number of unexpected observations that emerged during the experimentation campaigns. Among them, we could point out the fact that the asymmetrical antenna characteristics and brand design do have a strong impact on the communication capabilities of the nodes; for example, some nodes are good emitters, while other are better receivers. Such an asymmetry may be used to fine tune the graph representation of the network.

Thirdly, we investigate the differences between the traditional quantification of a contact and the model extracted from our measurements campaigns. Among other results, we reveal that when considering an adaptive throughput according to the distance between two devices, the long-distance data-exchange makes up more than 50% of the total data exchanged

in the entire network. We propose a tool to extract from mobility datasets the volume of data obtained, based on specific contact quantification strategies.

This thesis contributes to moving forward several aspects of the literature in regards to link utility and contact networks to accurately characterize D2D data exchange. Future works will focus on the asymmetrical characteristics of D2D communications revealed in this thesis to precisely model interactions.

Acknowledgments

Before getting involved into one, grasping the intricacies of being Ph.D. student is quite tricky. As a naive student, I perceived teachers as ever-lasting fountains of knowledge. During my first week, I asked my supervisors a technical question, to which they answered: "We don't know". Just like that, I transitioned from student to an associate in a heartbeat. It is as terrifying as it is exhilarating, and I would not have had it any other way.

First and foremost, I would like to thank Marcelo, Vania, and Farid for giving me this opportunity. Thanks to them, and the ANRT under the CIFRE fellowship, I had the privilege to evolve in both an industrial setting and an academical environment simultaneously, giving me a broad spectrum perspective of research as a whole.

Marcelo has to be one of the most stimulating people I have ever had the privilege to meet. His ability to motivate someone, and while always focusing on the silver lining, is unmatched. His overall positivity and humanity make working with him an absolute delight. He somehow managed to remain involved in my work without ever interfering with it, which is in itself a feat. I could not think of a better person to be supervised by, and I wholeheartedly believe I grew as a person as a consequence of our discussions, even casual ones. He always did the extra mile to help me, I'll forever be grateful for all the work he put in.

I want to thank Vania for making me feel at home at Thales. His input and feedback were critical to the success of this thesis. His industrial expertise greatly helped me anchor my thesis in a pragmatical fashion. I'm grateful for the fact that he chose to manage me despite his numerous responsibilities, and for the fact that always dedicated time for me when I needed guidance.

Last but absolutely not least, I would like to thank Farid for everything I put him through. And vice versa. They say honesty is the best policy, well I cannot think of a better person to embody this adage. His truth-seeking spirit made me thrive into the researcher I am today. He sees beyond silver tongues, for being well-spoken is not an excuse to shirk from scientific responsibilities. I'm aware this acknowledgement was as discombobulated as can be, as I know you enjoy deciphering my rambles.

I want to thank the examiners, Aline Carneiro Viana and Pierre Sens, for their continuous feedback and constructive criticism as members of the follow-up committee and members of the jury. I would also like to thank Katia Jaffres-Runser and Fabrice Valois for dedicating a significant amount of their time to review my Ph.D., and providing critical feedback on my work. Both examiners and reviewers undoubtedly helped elevate this thesis.

I want to thank all of my colleagues at Thales: Adrien, Agathe, Alan, Azharia, Bruno,

Dallal, Damien, Ehsan, Filippo, Geoffroy, Hicham, Kévin, Mathias, Marie-Noëlle, Matthieu, Maxime, Mehdi, Raphael, Sadia, Thomas. I truly enjoyed your company throughout these years and countless (morning) coffee breaks. I still do not like coffee nor mornings, but you guys made both bearable!

In particular, I'd like to thank Damien for pretty much being the dream carpooling buddy. From discussing computer networks to rap music, you're probably one of the few people with the superpower to make commuting to work actually fun. I'd like to also thank Thomas Aubourg for his direct help with the spectrum analyzer configuration, which helped me kick-off this thesis. I'd like to state that I also loved nerding-out, online or in real-life, with you two. I'd like to thank Mathias for his positivity and vital help for the field measurements. Sorry for the hours standing still holding a phone in the name of science! A special thanks to Adrien for our insightful walks back home.

I want to thank all of the tenured academic staff of NPA for their feedback, friendliness, and education they provided me throughout this thesis and my whole studies. A special thank you to Benjamin Baron, with whom this thesis would probably have never happened. Thanks to Salah, Yackolley, Bu, Pedro, and good luck to the ones who did not graduate yet!

In personal regards, I first would like to thank Rebecca for putting up with me during these 3 years. Your continuous support throughout these 3 years meant the world to me. Not everyone would have had the patience of doing what we did, thank you for everything.

To my best friends without whom I wouldn't be whole, Quentin, Romain, Charlène, Julieng, Sacha, and Thomas Lacheny: thank you. It would require a thesis of its own to explain how much you guys uplifted me, and I owe you more than I can ever repay.

I'd also like to thank the geek squad I've enjoyed relaxing with (or not) over the years: Esmaa, Riyad, Bastien, Mehdi, Adrien, Brice.

I want to thank my family for their unconditional support, Adèle, Andréa, Anne-Christine, Céline, Agathe, Sidoine, my godfather Pascal, Zamé, Mamie, GianCarlo, Franca.

Mom and Dad, thanks for always being here. You're the best. Always.

Contents

Listing of figures

1 Introduction

Over the last decade, smartphones and other connected devices have become a fundamental part of our lives. In 2017, there was an estimated total of 8.6 billion connected devices throughout the world and mobile data consumption grew 17-fold between 2012 and 2017, with an expected growth of seven-fold by 2022 [26, 27]. The consequence was a humongous growth of mobile traffic, ultimately leading to a saturation of the cellular spectrum.

Fortunately, by 2017, more than half of the total mobile traffic was offloaded from traditional cellular base-stations to alternative channels such as Wi-Fi or femtocells [27]. So far, this choice has been user-based, i.e., driven by the user's willingness to avoid using her mobile data-plan when unnecessary. The upcoming generation of mobile networks aims to push offloading solutions even further, by enabling new technologies to uncongest the network without the direct involvement of users.

Device-to-device (D2D) communication is a paradigm which states that devices may exchange data directly without having to go through an infrastructure. The idea is to make use of users' co-locality and mobility as a means to save bandwidth [55]. The very principle itself is not new – as a matter of fact, D2D communications were industrially standardized back in 2015 with the 12th release of the 3GPP under the LTE specifications as *proximity services* (ProSe) [17] but were not implemented in industrial cellular networks, most likely because of lack of financial incentives for network providers at that time [65, 106]. With the immi-

nent deployment of the 5G standard [40], several innovations are being proposed for higher throughput, lower latency, and overall spectral efficiency, with D2D among them.

The forthcoming of D2D communications creates an array of new possibilities. For instance, while its most evident usage would be to opportunistically offload mobile traffic [4, 82, 87, 115], it can also leverage the proximity of users to disseminate data to specific population [78], which is typically a prime use case for targeted advertisements [90]. Device-to-device brings a new form of traffic, which is not inherently based on traditional user-requested information but instead assumed to be of interest based on its neighbors or social ties [102]. D2D communications could also be benefitial in sparsely populated areas, since it can extend the range of cellular communications by leveraging nodes at the edge of the cellular range who act as relays for people out of conventional cellular coverage [55]. In the event of a catastrophic failure of the cellular network (e.g., natural disaster), devices could autonomously create a multi-hop network as a back-up emergency network [74, 85, 88]. This use case was commonly explored in the form of MANETs, but D2D communications may use licensed cellular spectrum, therefore bringing new possibilities [2, 3, 34, 75].

Device-to-device communications are likely to become a fundamental part of mobile communications because of their potential ability to bring services closer to the user, for nearly a fraction of the cost a regular operator would spend using traditional cellular communications. While its advantages are obvious, there are still several issues left open and the goal of this thesis is to tackle some of them, as we motivate and explain in the following.

1.1 Problem statement and positioning

The public recently took an interest in device-to-device communications when its real-life applicability has been demonstrated during protests [61]. D2D usage notably showcased its capacity to circumvent usual cellular communication's shortcomings, such as censorship or saturation in hyper-densely populated areas. While its efficiency in relaying text messages has been proven, both theoretically and practically, questions arise when considering other kinds of content such as large files. Nowadays, large scale dynamic networks are becoming a reality through the ubiquity of smartphones (or D2D User Equipment, DUE), along with IoT, and soon with vehicular networks, but the ever-increasing scale of such networks could be detrimental to conclusions previously drawn in the literature.

Let us take MANET as an example, where traditionally all nodes communicate under a single connected component. In reality, it is highly unrealistic to assume that an entire city forms a single connected component, thus resulting in multiple partitioned networks [92].

The mobile network research community then considered delay-tolerant networks (DTN) as a means to model direct interactions with multiple intermittently-connected components (which involves disruptions). In DTNs, no contemporaneous source-destination paths are assumed to exist. Wireless links are known to be volatile, since high individual mobility and interference may cause wireless links to lose connectivity at any time. Additionally, due to the heterogeneous characteristics of the DUEs, it is a common practice to have a different maximum communication range for each node. Last but not least, the mobility is not seen as an impediment, but rather as a method to carry and forward information to sparsely connected nodes [47].

With the DTN paradigm, several solutions have been proposed to alleviate typical issues from cellular networks, through the inherent usage of D2D communications, notably in offloading and data dissemination domains. The majority of the efforts were concentrated on achieving a high delivery ratio, short delay, and minimizing delivery costs [35]. A plethora of routing techniques and contact analysis have been proposed for DTNs [51, 119]. Nevertheless, when taking a closer look at actual device-to-device links inside a DTN context, we notice unusual gaps in the literature. The following paragraphs describe several problems we found most critical to address.

1.1.1 Centrality in a D2D network

Centrality metrics are widely used as a tool to route traffic in a communication network [96], with a wide range of applications in DTNs [33, 52]. However, when considering a D2D communication context, obtaining the traditional centralities of nodes requires full or at least partial knowledge of the topological structure of the network at all times. Due to the volatile nature of connected components in DTNs, computing a global centrality such as the betweenness is a hard task [71]. As an effort to minimize the computational time required to obtain centralities, the community has proposed several approximation methods, especially when it comes to the betweenness centrality [7, 19, 89].

A value of centrality can only be interpreted based on the aspect it aims to measure for

a given context. In other words, most centralities were initially coined for the purpose of analyzing static social networks. Even though they were proven to be useful in other domains, notably in DTNs/MANETs, it does not necessarily mean that they fit well such contexts. As such, re-defining centralities to tune the centrality measurement to the context is also a valid approach taken by the networking community [72], but not exclusively. For example, to emancipate from the static graph context, the *dynamic* betweenness centrality is therefore not calculated based on the available paths but on the journeys, meaning series of paths that consequently existed during sequential time-epochs [22]. The problem with this definition of dynamic centrality is that a global knowledge of the graph is still required. Furthermore, we need to be aware of all past, present, and potentially future edges existing between all vertices. As an attempt to alleviate the need for global network knowledge, the notion of ego-network was introduced [32]. To compute the ego version of a centrality, a node only takes into consideration its direct neighbors and the edges between these neighbors. Because of the high correlation between traditional centralities and ego centralities, it can be used as a way to estimate the real centrality of a node with only a small percentage of the computational cost [73]. Some variants propose to consider two-hop instead of one-hop neighborhood [59].

This thesis revolves around mobile networks and not pure complex networks or theoretical routing algorithms. While centralities are undeniably useful for routing algorithms, the actual communication requirement between the vertices (i.e., DUEs) to be aware of their neighboring nodes (i.e., to establish an edge) is frequently overlooked. Each communication comes with a cost since mobile nodes have a finite amount of battery life, they cannot afford to waste any energy. This, in a real-life setting, translates to another problem which relates to the topological awareness of nodes: maintaining up-to-date knowledge of the graph for all the nodes/vertices inside a connected component is yet another open challenge. As a matter of fact, one of the biggest problems around the topological awareness of nodes in DTNs is the latency required for all nodes to be notified about each topological change. In other words, on top of centralities (or alternatives definitions of it) being complex to compute on a large number of dynamically changing graphs, obtaining the actual graph itself is already a hardship.

In summary, obtaining centralities is an important component to efficiently route data in a network, but hardly applies to a dynamic context.

Problem 1

Is there a method to obtain a reasonably good estimate of the centrality of real-life DUEs without having to compute it every single time we need it?

1.1.2 Value of a D2D link

DTNs are modeled as graphs, where a vertex always represents a DUE, but edges have different definitions. In some works, they consider a time-varying graph where an edge signifies the presence of a functional wireless link between the two DUEs [93, 99]. A more complex perspective arises when considering contact graphs, in which the edges are weighed as a way to rate the importance of the link with respect to others. For instance, the weight of the edge could represent the number of consecutive time-epochs during which the two nodes had a functioning wireless link [52, 53]. Some suggest multiple weights for the same edges as a solution to compute shortest-paths based on different criteria, for instance, Shevade *et al*. [95] propose to weight link according to average available bandwidth, average delivery delay, and variance delivery delay. The common point between these solutions is that they all assign static values to links.

Problem 2

Current static metrics do not suffice to capture the potentiality of a link between two DUEs. We lack a method to quantify the intrinsic value of a D2D link to, in the long run, understand D2D data exchange opportunities on the macroscopic scale.

1.1.3 Fixed data-rate model

A contact is the continuous amount of time during which two devices are able to operate a valid D2D link. In the DTN literature, it is considered that a contact starts at the very instant when two DUEs are within communication range of each other [58]. A threshold distance, which aims to model the maximum communication range, is set according to the environ-

ment, needs of the experiment, specifications of the antennas, and so on. In this approach, two DUEs exchange data at a fixed throughput for the entire duration of the contact.

In reality, the distance between the nodes inevitably affects the performance of the link, besides other factors such as the presence of obstacles and interference from nearby communications [6]. In this thesis, we focus on the influence of the distance on the quality of a link.

Problem 3

Real-life wireless links are directly affected by the distance separating the devices. If the data rate is fixed, as suggested in the literature, can we actually obtain a realistic estimate of the volume of data exchanged during a contact?

1.2 Contributions and thesis outline

The above problems are all, to some extent, related to D2D performance issue. We will give the necessary background on a per-chapter basis, to exhibit aspects which have not been addressed by the community. In this thesis, we propose to address these issues from a pragmatical yet original point of view.

We now outline each chapter and their respective contributions. The thesis's workflow is briefly summarized in Figure 1.1.

1.2.1 Spatial centrality (Chapter 2)

First, we take a look at contact graphs from a theoretical standpoint, to familiarize ourselves with the concept and have a broad spectrum idea of recurrent issues which are found in the literature. Doing so, we notably observe the high computational complexity of commonly used graph metrics. These metrics are typically useful to understand the global (or local) importance of a node within a graph structure. This "importance" may be defined in many different ways, but the core idea is generally the same: the importance depends on how central a node is located in regards to the rest of the graph. Thus, this metric is commonly referred to as *centrality*. When considering static graphs, even large ones, approximating the centrality

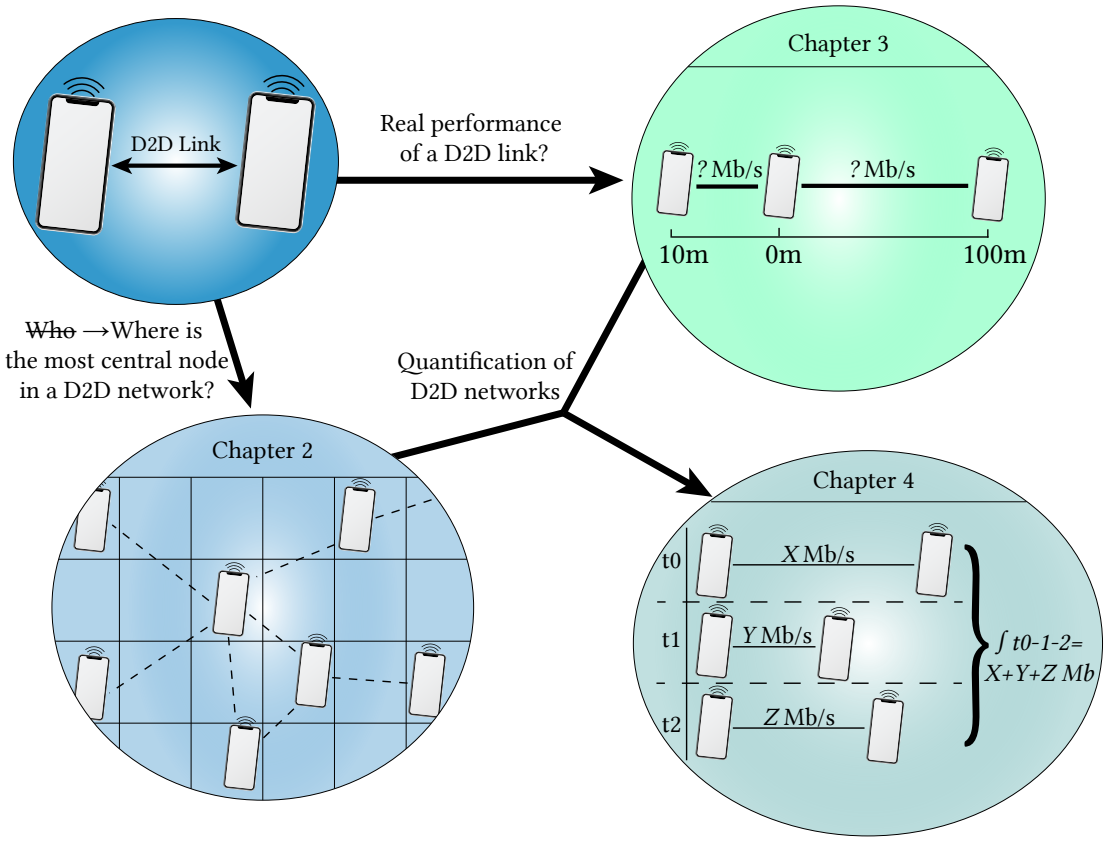


Figure 1.1: Visual summary of the investigative workflow of the thesis.

values of nodes is considered a viable option. To this extent, numerous approximation algorithms have been proposed by the community. However, in a D2D scenario, since the nodes are highly mobile, the centrality of nodes is likely to vary accordingly.

Since the network continually evolves, the computational time becomes a critical feature. The central question we address is: *is it possible to compute the centrality of nodes at the pace of the topological changes*? This sole question spawns other problems, such as *how frequent do the changes have to be in order to be considered unfeasible*, or *are approximation strategies accurate enough to capture topological changes?*

Let us make the hypothesis that approximation algorithms converge fast enough and accurately enough to find the most important individual in the entire D2D network, at all times. From a purely theoretical standpoint, this would solve our issue, but in reality, several factors

weigh in. First, in a D2D context, the construction of the graph is not instantaneous as it requires numerous message exchanges between all of the nodes of the actual graph. Even if we considered a scenario where base towers could roughly assume that all nodes within their cellular reach can communicate together, thus simplifying the contact graph construction process, there would still be a non-negligible latency issue. Indeed, since contact graphs may span over several base stations, it still requires inter-BS communications, thus having a latency issue to maintain up-to-date graphs. Second, from a sustained development as well as an economic perspective, the computational cost of maintaining up-to-date centralities, possibly at all times, may far exceed its benefits.

Instead of traditionally investigating contact graphs as mentioned above, we decided to bypass these questions entirely by taking an original approach and, to the best of our knowledge, never researched in the literature. Considering the D2D context of our study, nodes (i.e., mobile users) are in fact individuals carrying devices, and are roaming through space. The movements of users in space are arguably never random since not only they must have a destination, but the movements are also restricted by the underlying topography (e.g., a pedestrian must walk on curbs, a car must follow roads).

Therefore, the centrality of users could be directly correlated with their geographical location. We seek to estimate centralities of nodes, and by construction of contact graphs, using this intuitive observation. Considering a reasonable learning period, and the fact that day-to-day mobility does not change significantly, we could approximate the centrality of a node by merely looking up its position in space, thus mitigating the perpetual computation issue, complexity issue, and global network knowledge issue.

Related publications:

- C. Bertier, F. Benbadis, M. Dias de Amorim, and V. Conan, "Centrality Maps for Moving Nodes", *International Conference on Complex Networks and Their Applications* (Complex Networks), Cambridge, United Kingdom, 2018.
- C. Bertier, F. Benbadis, M. Dias de Amorim, and V. Conan, "Dis moi où tu es, je te dirai ce que tu vaux", *20èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications* (Algotel), Roscoff, France, 2018.

1.2.2 Direct link throughput modeling (Chapter 3)

We treat in Chapter 2 a theoretical issue (estimation of centralities) following a down-to-earth approach. In Chapter 3, we go a step further by going in-depth into what direct links are, and what they imply as of today's hardware and software. In the literature, ad hoc networks were initially considered in a context where the data transferred between two devices would be very small, in the order of a few kilobytes. As such, the majority of the works and algorithms developed to spread data into a contact network would revolve around the principle that once two nodes are in contact, i.e., when a wireless link exists, the transfer would automatically succeed.

Nowadays, mobile video traffic consumes the majority of the cellular bandwidth [27]; thus the community has actively considered video sharing within a D2D context to avoid cellular bandwidth to be wasted [9, 104]. Considering the size of a video file, along with the ever-increasing quality of smart-phones displays (therefore also increasing the size of video files), we cannot consider that a transfer instantly succeeds simply because two devices are in contact.

The next logical step would be to consider the data transfer rate between D2D devices to assert the success or failure of the file transfer. To our surprise, we found very little work done on file transfer between D2D devices, and more specifically on accurate estimation of bit rates between D2D devices. Additionally, the literature mostly consists of somewhat outdated work, considering the evolution of smartphones over the last decade.

In order to be able to assess the transfer capacities of smartphones, we propose to investigate the current state of D2D communications in real smartphones, to ultimately model their D2D links bit-rate accurately.

Related publication:

- C. Bertier, F. Benbadis, M. Dias de Amorim, and V. Conan, "Modeling Realistic Bit Rates of D2D Communications between Android Devices", *ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems* (MSWiM), Miami, Florida, USA, 2019.

1.2.3 Quantifying realistic contact capacities (Chapter 4)

If we recall Chapter 2, where we analyzed the centrality of mobile nodes, one of the motivations behind the work was to pinpoint nodes with high data diffusion capacity. However, in a wireless graph, a link solely exists if devices are within communication distance; otherwise, it does not. This binary perspective of D2D links highly conceals the reality behind such connections. For instance, it feels natural to imagine that nodes close to each other will effortlessly transfer data to one another. In other words, the link quality of close-by nodes is high, while faraway nodes have poor link quality.

In the problem statement, we mentioned that it is common to assume that the throughput between two nodes is constant (fixed) if it reaches a *reasonably* close distance. In this chapter, we tackle this issue by showing that using a constant value for the throughput during a contact may bias the results, whereas an adaptive throughput according to the distance between the nodes helps to obtain a more accurate characterization of the contacts. This chapter does not aim to advocate for an adaptive model (as opposed to fixed model) but instead focuses on characterizing the differences between the two.

Given a context where the geographical distance between two nodes is known, which is common if devices have their GPS location enabled, we can utilize our newly established goodput-to-distance model from Chapter 3 to estimate the instantaneous throughput between two nodes. Thus, by regularly sampling the distance between two nodes, we can estimate the total volume of data potentially exchanged during the contact, which we designate as the contact capacity.

Of course, parameters which we used to establish our distance-to-throughput model are not valid under every circumstance; it solely fits best the scenario under which we ran our experimental procedure. As an effort to generalize these results, we notably investigate how changing the throughput to distance estimation affects the contact capacity. Additionally, we provide capacity observations by varying the propagation conditions to fit different environments. By doing so, we also investigate how the environment in which devices roam affects the contact network.

Last but not least, we notably propose an open-source library* which takes as an input a mobility trace (e.g., GPS) and outputs all contacts which are uniquely identified, along

*https://github.com/Bertier/OpportunistiKapacity

with the duration of the contact and the contact capacity. Its functioning is explained in Appendix A, and we are hopeful that this incentive will push other researchers to compare our results to theirs.

Related publication:

- C. Bertier, F. Benbadis, M. Dias de Amorim, and V. Conan, "Computing realistic and adaptive capacity of D2D contacts", *International Symposium on a World of Wireless, Mobile, and Multimedia Networks* (WoWMoM), Washington, DC, USA, 2019.

I am a better poet than scientist.

Claude Shannon

2

Centrality estimation through spatial positioning

Base-stations taking sophisticated decisions such as offloading traffic to an individual carrying a smartphone (or a car) with device-to-device capabilities will soon become a reality. While there are several different barriers (e.g., security of information, battery consumption issues, legal complexities for information tractability) we found the decisional one to be of particular interest. Typically, the community has highlighted that a base station should carefully decide whether it should decide to offload, and to whom [83, 94].

A popular solution consists of sharing, when the conditions allow it, an information to an individual with proficient relaying capacity [101, 109]. Then, the chosen individual can share the data with its entourage through direct communications. To model relationships between individuals and finding the best-suited relay candidate, let us visualize this by picturing a graph. In this graph, a vertex represents a user carrying a mobile device, and an edge between two nodes exists if the devices carried by the two individuals are within wireless communication range of each other. Throughout this thesis, we label these graphs as *contact graphs*, because they represent all wireless contacts at a specific instant. Using a contact graph, we can compute the relative importance of nodes in order to find the node with the best relaying

capacity. We want to be able to compute the importance as a way to obtain the best-suited candidate to disseminate data.

2.1 Intuition

A *centrality* is a metric that measures the importance of a node in a graph by observing how much this node is central, compared to the rest of the graph. Several definitions of the centrality of a node in a graph exist; some are very straightforward to compute, such as the degree, and others more require more time and resources like the closeness or the betweenness. The more elaborate ones are useful and relevant for understanding technological, social, and biological networks, but have not been designed for large dynamic networks, because of the time complexity of their calculation [56]. Within the context of D2D contact graphs, knowing the centrality of nodes is paramount for mobile networks structural analysis [114] and is of particular importance for D2D offloading applications [98, 107].

Maintaining up-to-date centrality values at all times in such unstable networks might be unfeasible for two reasons. Firstly, several studies have proposed approximations to some centrality metrics [7, 89] though these solutions allow sufficient approximations, their computation time may still be too long compared to the speed at which the network evolves. In other words, in a real-time situation, the value of the centrality may be outdated when its calculation is completed and, as a consequence, no longer relevant to the current network [18]. Secondly, the network has to be notified of each topological change (i.e., a created or deleted link between two nodes). As such, there is a period of latency during which nodes may have an inconsistent view of the graph, thus resulting in an erroneous computation of the centralities of nodes depending on how much the update has spread.

We adopt a radically different approach to estimating the importance of a node in such dynamic topologies. We consider that all nodes represent physical mobile devices which are bound to spatial coordinates, such as GPS ones [36]. For instance, the coordinates could represent smartphones carried by pedestrians or vehicles with wireless communication capacity (as found in VANETs). Our idea relies on the observation that, in many scenarios, it is not necessary to know the exact numerical centrality value of all nodes of the network but rather to know which nodes are the most and least central [66, 79].

It is also based on an intuition that the *structure of the contact graph is shaped by the con-*

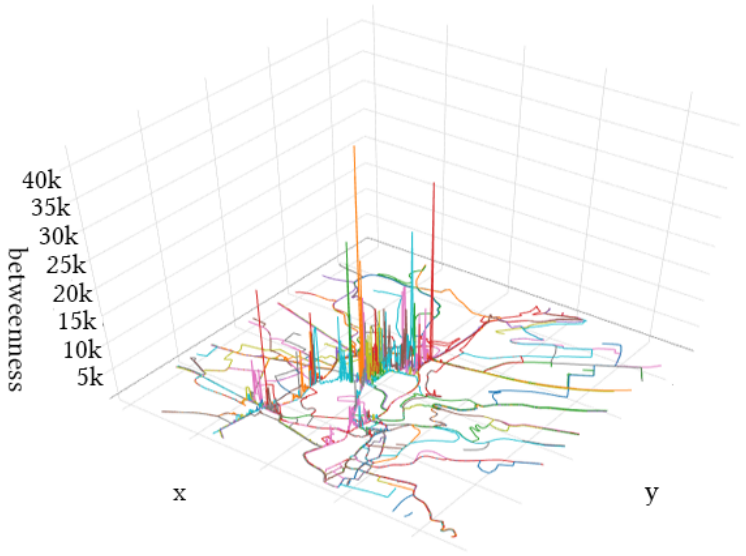


Figure 2.1: Betweenness centrality value of a 100 randomly-chosen cars in Luxembourg based on a V2V contact graph. Each color represents a different car.

straints of the underlying geographic area. Thus, by construction, nodes occupying some areas are likely to have higher centrality than nodes occupying other areas. This intuition comes from the observation of Figure 2.1, where we plot the paths of 100 randomly chosen cars (each color represents a car) in downtown Luxembourg during the morning rush hour (from 6 a.m. to 11 a.m.)*. The x-axis and y-axis represent the geographic position of a vehicle over time, and the z-axis represents betweenness centrality† of each car, according to the contact graph formed by the vehicles. Note that in the city center, which corresponds to the middle of the figure, there are surges of betweenness centrality. This central area, holding very high betweenness centrality, corresponds in reality to a very popular boulevard connecting several busy streets during this time of the day. In other words, vehicles in this area at this hour are likely to have a high betweenness centrality, whereas cars in the outskirts of the city tend to have low values. To the best of our knowledge, and despite this intuitive observation, no other work uses the geographic positions of nodes as a way to estimate their centrality.

*Though only 100 cars are shown (hence 100 3d-curves), there are at least a thousand cars simultaneously roaming the city during the day. More details are found in Section 2.4.1

†The betweenness centrality is the number of shortest paths going through a node. Its formal definition is found in Section 2.4.2.

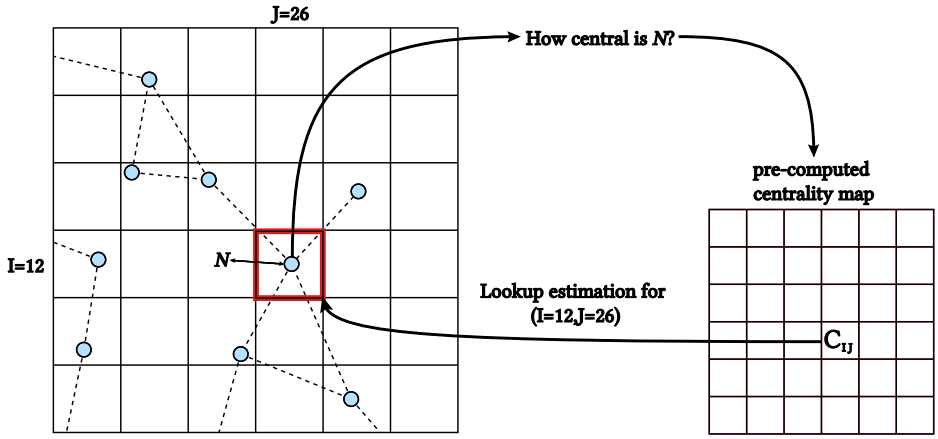


Figure 2.2: General centrality maps idea. The left part of the figure represents a contact graph, with a grid superimposed on top. To estimate how central node N is, we look-up a pre-established table based on the geographical coordinates of the node.

We summarize how our idea leverages the geographical position of nodes to estimate their centrality in Figure 2.2. On the left side of the figure, we have nodes and edges, representing contact graphs, with each node being placed at its spatial coordinates. On top of these graphs, we superimposed a grid with each square of the grid associated with a I,J index. Let us assume we need to estimate how central a particular node is, for instance the node N inside of square $I=12,J=26$, highlighted in red. Instead of going through the traditional centrality calculation process, we propose to estimate how central the node is by merely consulting a pre-established centrality table. The table, shown on the right side of the figure, contains the centrality estimation for all squares of the grid. The methodology will be thoroughly detailed in Section 2.3. We argue that the centrality of the geographical area (here, a square) can be used to estimate the centrality of the nodes lying in the same geographical area.

Our idea in this chapter involves a two-step process. We first observe during a learning period, at every moment of the day, the contact graph formed by the vehicles or pedestrians, compute the nodes' centralities, and associate these centralities to the geographic positions where they were measured. As such, we obtain a *centrality map* of the region which is fortunately stable over time, mainly in areas where the centrality is very high or very low (which are centralities of particular interest). Then, for nodes evolving in a target area, we do not have to compute their centrality at all times – it suffices to refer to the centrality map to know how

central they are.

As a summary, the contributions of this chapter are:

- **Centrality map.** We present a method to create a geographical centrality database, by dividing the surface into squares and utilizing the centrality of nodes located inside the squares as a way to estimate the square's centrality.
- **Geo-centrality.** We propose a new way to identify quickly, and as fast as querying a database, the nodes with the highest centralities in large-scale dynamic networks. This method, based on the creation of centrality maps from data collected allows immediate access to an estimate of the centrality of a node based on its geographic coordinates only.
- **Application to real-world scenarios.** We evaluate our method in large-scale vehicular and pedestrian networks. We compare the current geographical centrality of a square to the real centrality of nodes living in this square. We observe that the estimated geographical values reflect well on nodes with the lowest and the highest centrality values, often regardless of the type of centrality metric.
- **Application to content distribution use case.** To measure the relevance of our centrality estimation method, we evaluate it as part of a content distribution protocol based on the centrality of the nodes and compare its performances with the same protocol using exact centralities calculated with classical methods. We find that our method gives results at least as good as those of conventional methods. The results we get with our method are sometimes even better than the methods that require more time and resources.

2.2 Related work

To the best of our knowledge, very little work has been done on the position of nodes inferring the centrality. The idea of a "geographical centrality" did exist, but with a meaning different from ours. In most works we found, the notion of centrality and geography was within the context of urban studies [12], such as the study of transportation networks (commuting services, airports) [49, 105] or street/road layout [31, 100]. These examples do not compare to

our work at all; in our case, dynamic graphs are constructed based on the geographical proximity of mobile nodes (e.g., humans or vehicles), whereas the works mentioned above study static graphs based on a more abstract notion of links (e.g., transport of goods/people or city's road topology), or direct street adjacency.

Computing or estimating centralities within a reasonable time is still an on-going issue, due to both the natural high-complexity of certain centralities, and the scale of the studied graphs. There have been numerous approximation solutions proposed [1, 7, 8, 29, 39, 89]. They take different approaches to the approximation issue, for instance, one solution consists of lowering the number of calculations by sampling the nodes [39, 64, 89], another solution consists of distributing the calculation to enhance the scaling [70]. An interesting approach, used within the MANET community, focused on considering an egocentric network, where the centrality is only calculated with its 1-hop neighbors [32, 73], and has proven to be an excellent approximation for the betweenness.

The closest idea to our work proposes to predict the centrality value of nodes based on the passed centrality values of the same nodes [60]. In the latter, authors study human contact networks, in which they demonstrate that the centrality value (betweenness, closeness, and degree) of nodes at a given time t is highly correlated with values of the same nodes at a future time $t + \tau$. They leverage this periodicity to obtain the actual value of a node based on its passed value. Nevertheless, none of the works we have found used the geographical location of mobile nodes as a solution to deduce the centrality.

Still, in the context of MANETs/VANETs, there have been several efforts to leverage the geographical (or topographical) position of cars to find the most suitable candidate to disseminate content as efficiently as possible. In SADV [112], the dissemination method is based on the awareness of intersections; if no node is found within communication range, and if the current holder of the information is in an intersection, the data is transferred to a static-node assumed to be existing in the middle of the intersection. Later, the node will forward the data to vehicles passing in the intersection. A similar idea is found in Throwboxes [117], where the authors state that knowing contact opportunity is necessary. In [116], a data-center tries to disseminate information (accident, traffic-jam) to vehicles within a pre-defined dissemination zone. The data is broadcasted on specific roads, knowing they intersect other roads of the area. To avoid using the data-center too often, they propose to use *relay and broadcast stations*, located at intersections, whose purpose is to copy the information and forward it to

other nodes. In all the examples above, it is assumed that intersections are the areas where the most opportunistic forwarding should happen, given the density and transient nature of the vehicles found inside of them. Additionally, numerous VANET algorithms make use of the topography to disseminate the content [108, 110].

The aforementioned works assume some correlation between the geographical position of a node (e.g., vehicle) and its ability to efficiently disseminate data. To our surprise, they have attempted to leverage the geographical position as a way to study contact graphs, and more specifically to potentially deduce a node's centrality. Lima *et al.* have proposed to redefine centralities to make them usable within a spatial context [67].

2.3 Our proposal: Geo-centrality

We are looking into a method to estimate the centrality of mobile nodes, in a contact graph. With this method, we aim to provide an estimate reasonably close to the real node's centrality without requiring as many calculations. We assign, based on the day of the week and the time of day, a centrality value to an area in space, and not a node.

Therefore, when we want to know the centrality of a node at a given moment, we refer to its position in space, and take in consideration the time of day as well as whether it is a weekday or holiday. Other features may be We utilize the fact that human mobility patterns are redundant, depending on the type of day [41], so that our results generalize to days with the same type. We first calculate a centrality map, a database that lists the centralities by point in space and by the time of day. This section explains our method for the centrality map calculation. Once a centrality map is obtained, using our centrality map for a node's centrality estimation is as easy and fast as looking up in a database.

2.3.1 Problem statement

We consider the case of a communication network, made up of mobile nodes whose centrality we want to know. Since the network is mobile, the centrality we wish to compute must be instantaneous and evolves with each change in the topology. At each moment, then, we consider a snapshot of our network and construct the graph whose vertices are the nodes. An edge between two nodes exists if the euclidean distance between them is less than Δ. From

this graph, we compute the centrality of every node. So far, this entire process has no novelty to it. The issue here is that the construction of the graph and, in particular, the calculation of the centrality of every single node is very greedy in terms of resources and calculation time, which makes it scale very poorly.

Another issue with this traditional graph centrality calculation is the fact that mobile nodes imply dynamic graphs. As such, even if we can obtain perfectly accurate centralities of all nodes in the snapshot, the centrality values of all nodes may be completely different at the next snapshot, even if a short time separates the two snapshots. The time between two topology changes is much shorter than the time required to calculate the centrality of all nodes. Thus, the result of the calculation of centrality is no longer valid once it converges, since the topology has changed.

We argue that the sole geographical position of a node may be enough to approximate its centrality. As such, the *Geo-Centrality*, noted *geo-c* in the rest of the chapter, is not a new centrality but a method to map a traditional node-based centrality onto a geographical area; in other words, *geo-c* turns geographical coordinates into a centrality estimation tool.

2.3.2 Centrality map calculation

Let us rely on Figure 2.3 to explain our method to calculate the *geo-c*, and by extension, the centrality map, the database storing the centrality estimations of all geographical positions.

On the left side of the figure, inside the dashed frame, we have:

- The temporal window W, this represents the temporal granularity of the centrality map,
- W is divided into 4 time-stamps, noted t_1 to t_4,
- A geographic area, represented at each instant t, divided into 9 squares,
- Up to 6 nodes, noted $n1$ to $n6$, living in the geographic area. These nodes may join and leave the area during W,
- a red square, noted S, for which we compute the geo-centrality for time window W,
- And $\gamma(t, n)$, our centrality estimated value for node n at time t.

We explain in this section how the geo-centrality of S is computed. The same method will be applied for all the squares of the considered area to obtain the complete centrality map.

Consider that we have a function γ, explained in details in the following subsection, which computes the centrality estimated value of node n at instant t. The γ function with the topology represented in Figure 2.3 gives the following results:

1. At t_1: $\gamma(t_1, n1) = 1$
2. At t_2: $\gamma(t_2, n1) = 1, \gamma(t_2, n_2) = 1$
3. At t_3: $\gamma(t_3, n3) = 1, \gamma(t_3, n_6) = 5$
4. At t_4: $\gamma(t_4, n5) = 5$

These estimations of nodes' centrality values living in square S during period W, including the four instants t_1 to t_4, are used to populate the array $\gamma(W, S)$, which represents the centrality estimated value of S during W. In our example, this array equals to $[1, 1, 1, 1, 5, 5]$ and represents the γ-values of square S during window W. This array gives the histogram of γ for S during W (right-hand side of Figure 2.3).

This histogram allows deducing the *geo-c* of S during W based on the γ with the highest probability, 1 in the example we depict here.

2.3.3 Calculation of γ

Our approach is centrality-agnostic but, for the sake of illustration, we consider the betweenness centrality in Figure 2.3. We evaluate, however, different centrality metrics in the evaluation section 2.5.

In dynamic graphs, the notion of high or low centrality values quickly becomes blurred because of the constant topological changes and a fluctuating number of nodes. This makes absolute values of centrality difficult to interpret [60]. A better approach is to rely on the relative centrality of the nodes by ranking them [16] in decreasing order of centralities. Nevertheless, because such a ranking is still too unstable, we make a step forward and consider the quantiles of ranks. We refer to this rank as the $\gamma(t, n)$.

Computing square S's geo-centrality during window W, noted *geo-c*(W, S), requires to obtain the $\gamma(t, n)$ of all nodes n inside of square S. A value of γ is always calculated on nodes

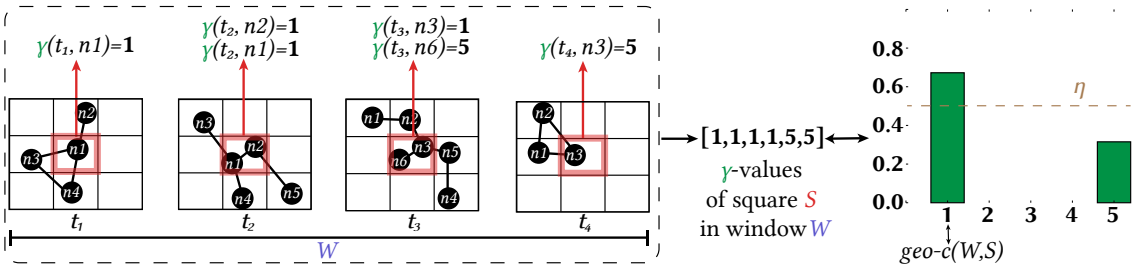


Figure 2.3: General methodology to create a geographical centrality. Here we have 4 different snapshots with different topologies (inside the dashed frame, on the left), and we focus on a single square S (highlighted in red). By analyzing the centrality values of nodes (function γ), we can establish a geographical centrality for S (*geo-c*).

that existed inside of S at specific snapshot $t \in W$. To clarify how to obtain the $\gamma(n, t)$ for all nodes n existing inside S at t, we go through the following steps:

1. We compute the centrality (betweenness centrality is considered in this figure) of all nodes at the specific snapshot $t \in W$. In Figure 2.3, at instant t_1, centrality of node $n1$ is equal to $\frac{2}{3}$ while centralities of $n2$, $n3$, and $n4$ are equal to 0. If the square were to be empty of nodes at snapshot t, we do note compute any γ.

2. We rank all nodes based on their computed centralities. In the considered figure, at time t_1, node $n1$ has the highest centrality and is then at rank 1. $n2$, $n3$, and $n4$ have the same centrality value (0) and have then rank 2.

3. We segment the ranks in N *quantiles*, with each *quantile* holding $\frac{100}{N}$ percent of the ranks. Without loss of generality, we fix $N = 5$. Hence, we have $\gamma : X \mapsto \{1, 2, 3, 4, 5\}$, where the first quantile corresponds to the top 20% of the ranks, the second quantile corresponds to the range 20-40%, and so on. In Figure 2.3, at instant t_1, we consider only node $n1$ since its the only one in square S. $\gamma(t_1, n1) = 1$ then.
   Note that because of this segmentation, we require at least 5 nodes with 5 different ranks so that no quantile is empty. We suppose this condition always true as our datasets always fulfil this requirement.

4. We repeat steps $1 - 3$ for every instant t of window W. We can see, for instance, that γ of nodes $n3$ and $n6$ are respectively equal to 1 and 5 at instant t_3. That means that $n3$ is among the top 20% values and $n6$ is among the lowest 20% values, resulting in $\gamma(t_3, n3) = 1$ and $\gamma(t_3, n6) = 5$.

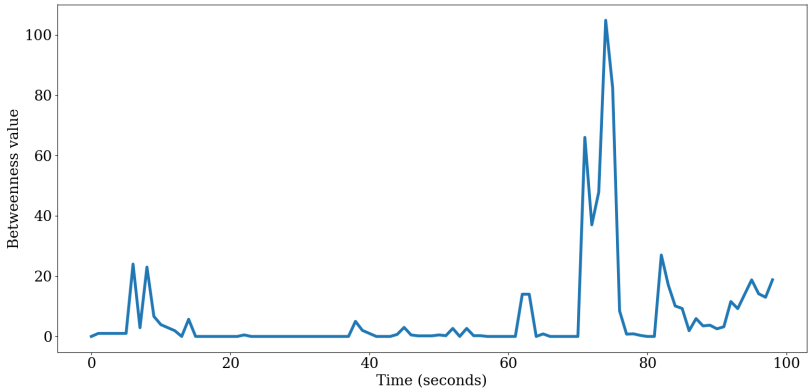


Figure 2.4: Betweenness value of a random node over time. The variations over time do not show an obvious pattern, making it complex to anticipate future values.

2.3.4 Distribution of γ

Now that we have the γ (quantiles) of the nodes within square S in the window, we aggregate these values to obtain the centrality of the square. In Figure 2.3, the γ of the nodes inside of S are [1,1,1,1,5,5].

The next step is to determine the probability distribution of these values, as represented on the right side of Figure 2.3. We note the probability density of a quantile q in square S during window W as $\delta_q(W,S)$; in our example, we have $\delta_1(W,S) = \frac{2}{3}$, $\delta_2(W,S) = 0$, and $\delta_5(W,S) = \frac{1}{3}$, and so on.

We aim to find squares with a representative quantile, i.e., a quantile with much higher probability than the others. To this end, we define a *stability threshold*; a square is considered as *stable* if one of its quantiles above η.

The *geo-c* of a square is defined only if the requirements below are met, otherwise we consider its *geo-c* to be *undefined*:

$$\textit{geo-c}(W,S) = q \leftrightarrow \exists! q, \delta_q(W,S) > \eta. \tag{2.1}$$

2.4 Applying centrality maps to vehicular networks

When observing the centrality value of a node, for instance in Figure 2.4 we observe the betweenness-centrality of a random node over 100 seconds, we noticed that values prior to time t were not evidently correlated with of the value at time t. In other words, "predicting" the current centrality value of a node based on its previous values does not seem straightforward.

Our solution offers to rely on the geographical position as a way to predict the current value of a node. As such, we must verify that the passed centrality values of nodes in a given area will help predict the current centrality values of nodes within this very area.

In short, the general idea is that we aim guess the present centrality of a node by looking at past *geo-c* of the square it falls in. For all time steps t spaced by τ:

1. We compute the geographic centralities *geo-c*(W_t, S) of all squares S of the map on the sliding window $W_t = [t, W + t[$.
2. We compute the γ of all nodes at the time $W + t + \tau$.
3. For each node n within a square S, we compare the true centrality quantile of the nodes, $\gamma(W + t + \tau, n)$ to the one predicted by its geographic position *geo-c*(W_t, S) when it exists. If both values are equal, we count the prediction as a success.

2.4.1 Experimental datasets

To verify whether nodes can be predicted from their geographical positions or not, we first and foremost require mobility datasets. These datasets consist of the pinpoint of a user at a specific time (e.g., through GPS). For contact graphs to be accurate at all times, which is mandatory since we need to obtain the exact centrality of all nodes at any given time, we require both fine precision and frequent updates in the dataset.

Unfortunately, we found no open-source real-life mobility traces with updates sufficently regular to ensure consistency in our graph calculation. As such, we will rely on the following peer-reviewed emulated vehicular datasets[‡]:

[‡]Ideally, having fine-grained and accurate datasets spanning over several days, perhaps months, would have grandly facilitated the study.

- LuST [28] covers 172 km^2 over the city of Luxembourg, with approximately 300,000 vehicles over 24 hours, typical workday.
- TapasCologne [103] covers 400km^2 over the city of Cologne, with approximately 700,000 vehicles over 24 hours, typical workday.

For both datasets, we focus on the city center, a surface of approximately 20 km^2 in both cases. We calculate the centrality of nodes (vehicles) of the contact graphs. To construct the graph, we consider that two vehicles form an edge if they are within a communication radius Δ of each other. In this work, we take $\Delta = 50$ meters. It corresponds to a relatively short distance in the context of vehicular networks, which is a reasonable assumption given our high-density urban environments.

These two vehicular datasets differ from one another in several aspects. The road layout of Luxembourg is more star-shaped than Cologne's, whereas Cologne has a shorter and more crowded rush-hour. In terms of graph dynamics, the Luxembourg dataset forms significant connected components in the same areas throughout the day, whereas in Cologne, they change over the time of day.

In our analysis, we set the following parameters: window $W = 60$s, time step $\tau = 10$s, and stability threshold $\eta = 0.5$. This value of η is chosen so that only one quantile may represent the geo-c of a square, since more than 50% of all γ values within it must belong to the same quantile.

2.4.2 Node centralities

There are numerous centrality metrics in the literature [37], but here we only focus on the most well-known in a networking context.

Degree centrality counts the number of direct links of a node. It provides a value of the local importance of the node in the graph.

Closeness centrality represents the average distance between a node and the rest of the graph. It needs to be computed over the entire connected component. Noting $d(v,u)$ as the distance (in terms of number of hops) between nodes v and u:

$$clos(v) = \sum_u \frac{n-1}{N-1}\frac{n-1}{d(u,v)}. \tag{2.2}$$

The $\frac{n-1}{N-1}$ factor aims to weight the metric according to the proportion of reachable nodes in the connected component of v, with n being the size of the connected component and N the total population.

Betweenness centrality gives the proportion of shortest-paths that go through node v. Noting σ_{sd} as the number of shortest-path between a source s and a destination d, and $\sigma_{sd}(v)$ as the number of these shortest-paths that go through node v:

$$bet(v) = \sum_{s \neq v} \sum_{d \neq s \neq v} \frac{\sigma_{sd}(v)}{\sigma_{sd}}. \tag{2.3}$$

Egobetweenness centrality is a local approximation for the betweenness centrality [73]. It follows the same formula as the betweenness of a node but limited to its 1-hop neighbors component.

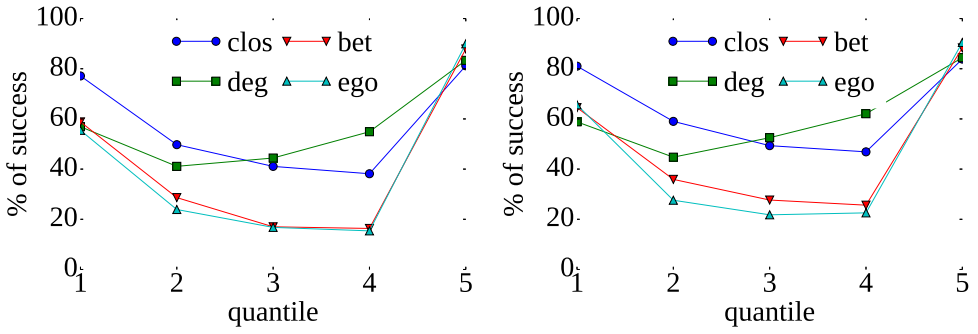
2.5 Results

Figure 2.5 presents the success rate of predicting the nodes centralities from their positions: γ of the node is compared to the *geo-c* value of the square it falls in (see Section 2.4), and results are given per quintile. Evaluations were carried out for both Luxembourg and Cologne datasets over the entire day-time period (from 5.30 am to 10 p.m.).

The overall pattern for both cities is strikingly similar. Prediction rates are best for the first and fifth quantiles: if the node is within the top 20% values or lowest 20% values, its node centrality is correctly provided from its position. This is true with 82 to 90% chances if it lies in the fifth quantile and between 60 to 80% chances if it lies in the first quantile. Note that betweenness and ego-betweenness are the only ones with prediction success rate lower than 50%.

2.5.1 Contact density

To understand better the success rate of each centrality, in Figure 2.6 we provide the number of predictions per quantile (in millions of prediction attempts), for all centralities, along with the proportion of successes (light green) and failures (dark red).



(a) Luxembourg **(b)** Cologne

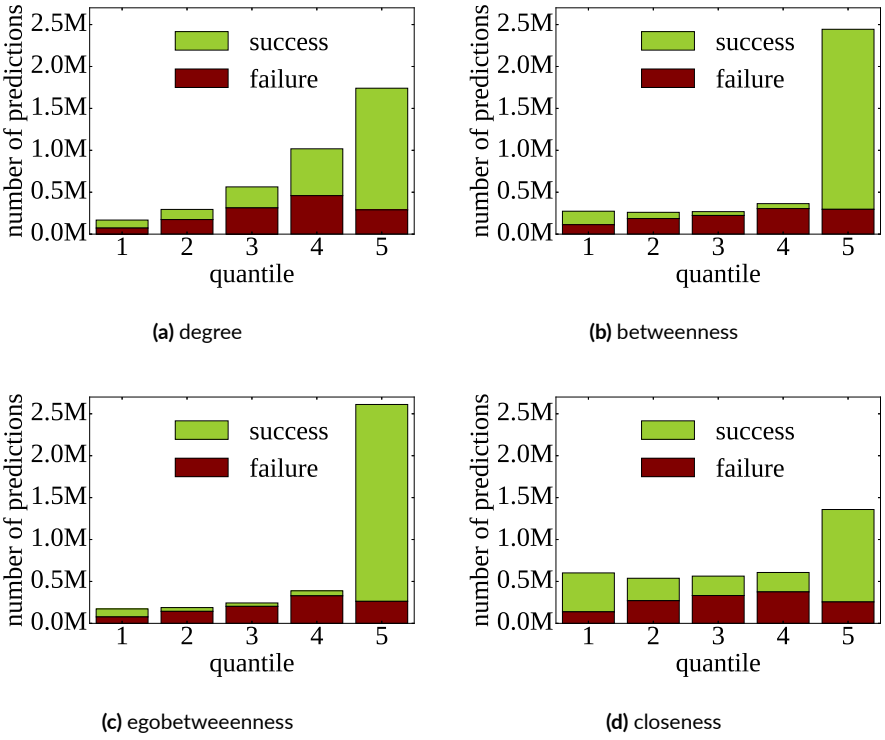
Figure 2.5: Success rate of node centrality prediction per quantile for degree (deg), closeness (clos), betweenness (bet) and ego-betweenness (ego) centralities.

Keep in mind that the number of nodes in each quantile may vary depending on the centrality metric. Indeed different centralities yield different behaviors in terms of value distribution. This is a pure graph property; for instance, the betweenness is known to have few very high valued nodes because of its flow-based nature, but nodes located even one hop away from the highest betweenness-valued node may have a value of zero.

Since we only take into account squares which have a defined *geo-c*; depending on the centrality, the same square at the same time may or may not have a defined *geo-c* value. This explains why the four centralities have different behaviors: for the betweenness (and ego-betweenness) most values are found in the 5th quantile, where nodes have low betweenness centrality values. The number of nodes within quintiles 1 to 4 is much smaller than those within the 5th one.

However, all the centralities reveal a similar pattern: the 5th quantile holds the largest number of predictable nodes (i.e., cars located in a square with a defined *geo-c* value) and this 5th quantile also yields the best prediction results (between 80-90% successes on all four centralities).

These squares with low *geo-c*, therefore, correspond to areas where cars have limited contact opportunities. The takeaway here is that we can easily predict low centrality values of nodes based on their positions, hence predicting a large proportion of the population.



(a) degree

(b) betweenness

(c) egobetweeenness

(d) closeness

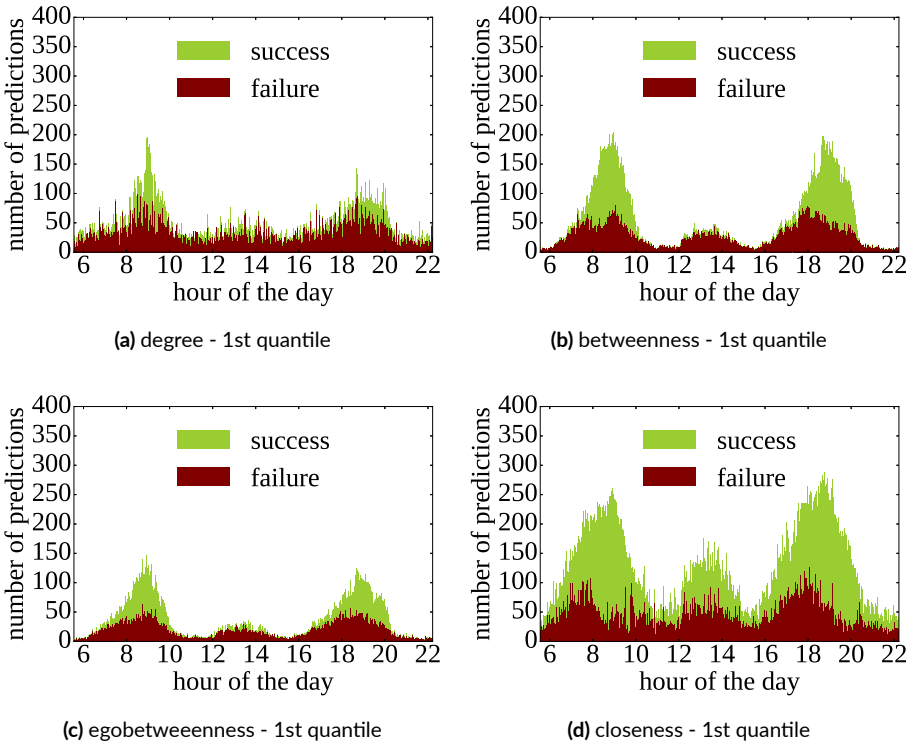
Figure 2.6: Number of node centrality predictions per quantile for the four centralities.

2.5.2 Evolution over time

In Figure 2.7, we show how the prediction success rate evolves over time, during a day-long period. We plot success and failure rates for the first quantile (i.e., nodes with centrality values among the top 20%). For all four centralities, the plot shows a pattern corresponding to three periods: the morning rush (from 6 a.m. to 11 a.m.), the noon break (12 p.m. to 3 p.m.), and the evening rush (4 p.m. to 8 p.m.).

The degree centrality (Figure 2.7a) shows a dented pattern. This means that success/failure patterns are not strongly time-correlated in the short term. Predicting degree centrality is very uneasy: there are no long periods where success outnumbers failures (but the total number of successes is still larger than failures).

For betweenness and ego-betweenness, represented in (Figures 2.7b and 2.7c, a better



(a) degree - 1st quantile

(b) betweenness - 1st quantile

(c) egobetweeenness - 1st quantile

(d) closeness - 1st quantile

Figure 2.7: Evolution of node centrality prediction (success and failure) over time, for the four centralities and the first quantile.

prediction is achieved for the rush hours, either in the morning or in the evening. At noon, it is difficult to obtain good predictions.

The closeness centrality (Figure 2.7d) leads to better and more regular prediction successes – it is significantly less sensitive to population density the majority of the time; no matter the population or hour of the day, there are more successes than failures. Observing these centralities over time helps to shed light on their behaviors. They respond differently to population density, with the closeness being the less sensitive, as well as providing an excellent prediction for the morning and evening rush-hour periods.

On Figure 2.8 we focus on the morning peak of the rush hour period; we plot the success rate of the closeness centrality prediction every 5 minutes from 8 a.m. to 10 a.m. Prediction, in this case, is very successful: there is at least 80% chance of successfully predicting the cor-

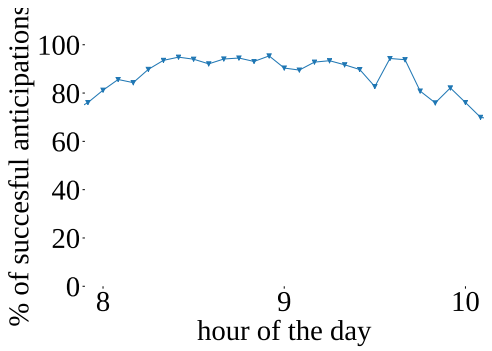


Figure 2.8: Successful predictions every 5 minutes during the morning rush-hour.

rect quantile, with a maximum of approximately 95% of nodes' quantiles correctly predicted. Even if centralities are sensitive to population density, as we have seen above, when we focus only on very dense scenarios, we can ensure a very high prediction success rate with our methodology. Such results *strongly encourage that creating a database matching a centrality to a particular time of day is feasible*.

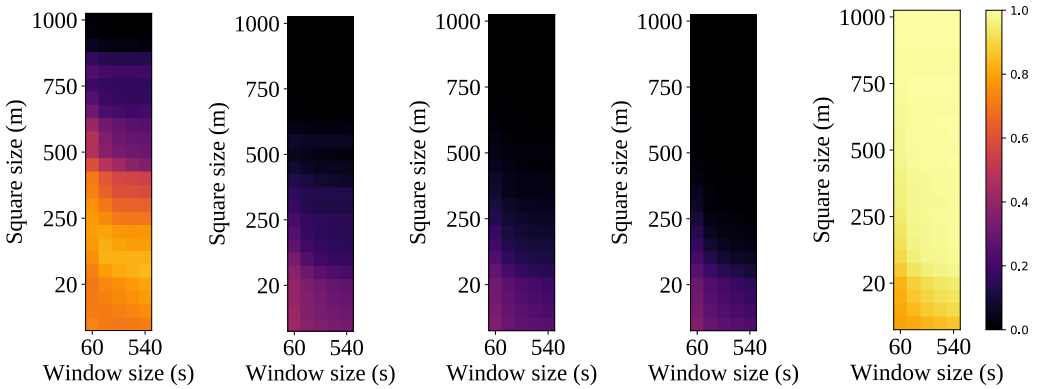
2.6 Influence of parameters

While we have previously determined the applicability of our method for short-term centrality estimation, the window size and square size parameters were set for simplicity's sake. Since our model has numerous parameters, we deemed necessary to discuss their influence on our results.

2.6.1 Success rate according to the window and square size

To properly understand the benefit or disadvantage for a given value of a parameter, let us consider the average prediction success rate of each quantile. To do so, we compute the mean prediction success rate over the entire dataset as an estimation of the global efficiency of the parameters used. In Figure 2.9, we present this mean success rate of predictions, when using the closeness centrality, while varying the window size and the square size.

Let us first observe the results for the first quantile (top 20% values), as shown in Figure 2.9a. When choosing a large square size, for instance of 1000m, we observe that the



(a) 1st quantile **(b)** 2nd quantile **(c)** 3rd quantile **(d)** 4th quantile **(e)** 5th quantile

Figure 2.9: Heatmap of mean prediction success according to window size and square size, per quantile, for the Luxembourg dataset.

success rate goes to zero. Attempting to predict the centrality of nodes in an area this large (1km) does not work as expected, for several reasons. The nodes inside the square will either be too diverse (belonging to different quantiles) hence making all squares' *geo-c* undefined, or, squares will have a defined value of *geo-c*=5 since nodes of the fifth quantile are count for the majority of values of the dataset. In any case, trying to find a vast zone where all nodes belong to the same first quantile is not possible. While choosing a minimal square value (e.g., 5 m) may seem like it will automatically yield the best results, we can observe that the smallest square size results in a success rate of 70-75 percent. When a square size of ≈50 meters is chosen, the success rate is at its highest, reaching 80-85% success. This observation is of particular importance because it reveals that a trade-off between a large square (too many different values) and a small square (not enough values) is the optimal solution.

Figures 2.9b,2.9c and 2.9d, respectively representing quantiles 2, 3 and 4, all seem nearly identical. Here, while the success rates presented are globally smaller than the ones in the first quantile, a clear pattern emerges: shorter window size (60 seconds) as well as smaller square size (5m), leads to the highest success rate.

Let us reflect on the implications of the latter observation. A small window infers that less information is accumulated for the geo-c's calculation. A smaller square also infers less information, because fewer nodes are likely to reside within its borders. Hence, the fact that

quantiles 2-3-4 have a higher success rate using both small window and small square size showcases the difficulty of accurately localizing nodes belonging to these quantiles, because not only is the *geo-c* is suitable for a very short duration, it is also only valid for very small areas. Either nodes from these quantiles are extremely mobile (thus hard to locate over time), or the nodes' centrality value constantly switches between quantile 2-3-4. In any case, an accurate prediction based on the *geo-c* is impractical.

Lastly, when observing the results of the fifth quantile in Figure 2.9e, we can observe a clear correlation between the two parameters. Contrary to quantiles 2-3-4, this time if the window is small (60 seconds), and the square size is also small (5m), then the success rate is at its lowest. Reciprocally, if the window is large and the square size is large, the success rate is the highest (≈ 0.95). We remind the reader that a given success rate must be interpreted within the context of its own quantile, thus while the "lowest success" rate of the fifth quantile is approximately 80 percent, it is still tremendously higher than the success rate of the other quantiles. Contrary to our observation for the first quantile, there is no trade-off to obtain for this quantile, the larger the square and window, the better.

In summary, nodes from quantile 5 (the least central) grandly benefit from larger windows and larger squares. This shows that such nodes are highly geographically correlated and zones in which they evolve are stable through time. On the opposite, nodes from quantiles 2-3-4, due to their instability, are very difficult to locate accurately. The influence of parameters on nodes from the quantile 1, the most central nodes, are a bit more complicated. To achieve the best prediction rate, tuning of parameters is required to enable 80 to 85% success rate.

We can only set a single square and window size for the entire anticipation process. One could potentially choose these parameters according to the quantile that one wishes to anticipate the most efficiently.

2.6.2 Influence of window on predictability rates

As a solution to help decide how parameters should be set in case one does not wish to achieve maximum success rate for a particular quantile, we propose to investigate the percentage of predictable nodes. A node is "predictable" at time t if the node is located within a square with a defined *geo-c* value at time t.

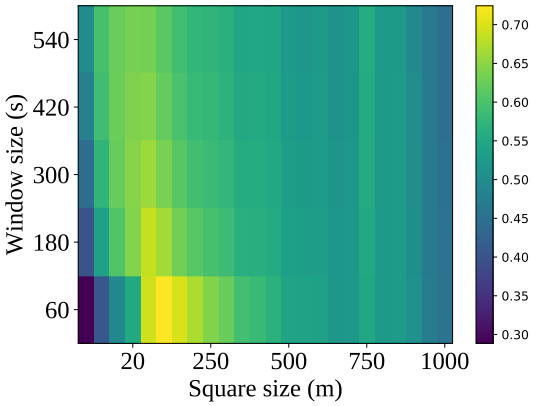


Figure 2.10: Mean percentage of predictable nodes, according to window size and square size.

One may believe that observing the number of predictable squares is a better-suited metric, however, doing so does not take into account the actual population inside the squares, and would strongly be biased by squares which have seen a single node for the duration of the entire window.

As such, in Figure 2.10, we observe the mean percentage of predictable nodes, according to window size and square size. First of all, when using enormous values such as 1000m squares and windows of 540 seconds, we reach a predictability rate of approximately 45% of nodes. When squares are this large, their *geo-c* is either undefined or is equal to the fifth quantile, since the majority of nodes are from the fifth quantile (as seen in Section 2.5) and realistically leave no other possibility.

Taking minimal values (5m and 60 seconds) is not a good solution either, since it only enables the prediction of nodes from quantiles 2-3-4, which represent a small part of the population, thus only yielding 30% predictability rate.

Still, on Figure 2.10, we can observe that the highest predictability rates are achieved by selecting values which benefit the prediction success rate of all quantiles. For instance, we observe that a small window size of 60 seconds and a reasonably sized square between 20 and 150 meters allow the predictability rate to reach up to 70%.

2.6.3 How to set window and square sizes

From these observations, we now advise two strategies to set the window and square size parameters. The first solution requires to select a "target quantile" beforehand, meaning a particular quantile of interest that one wishes to achieve maximum predictability onto. Then, tune the parameters to achieve the best success rate for the target quantile. The second solution, which requires no prior select, consists of merely choosing the value which results in the highest predictability rate.

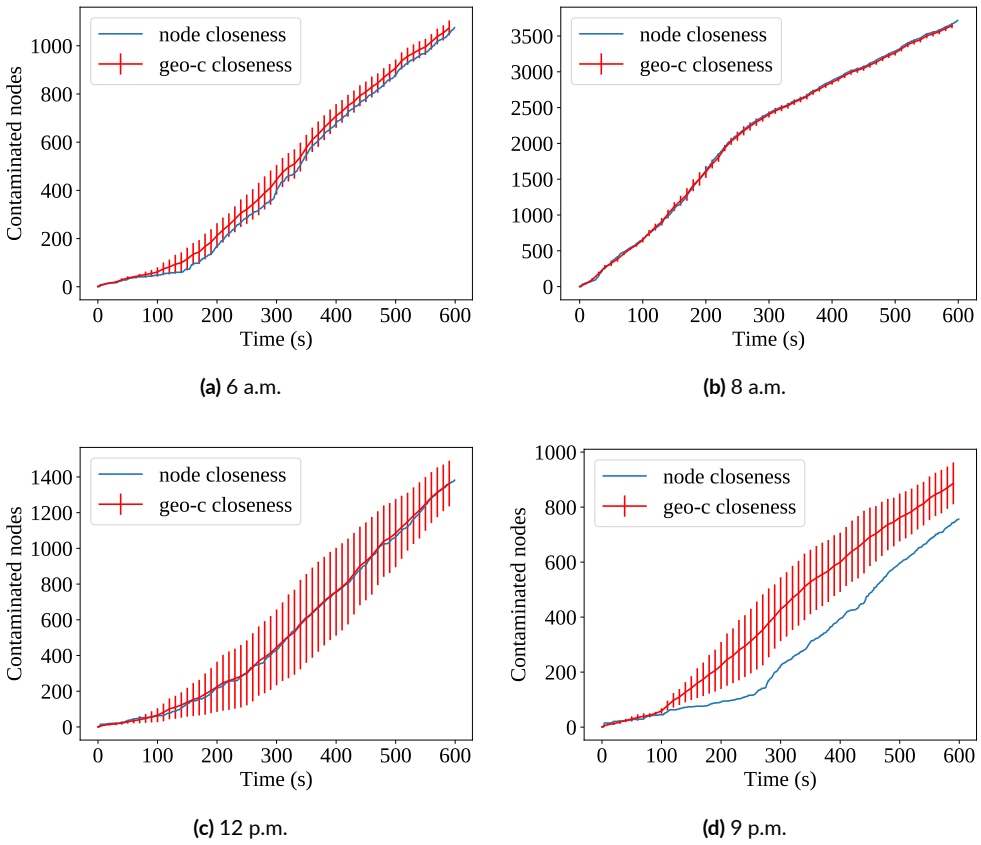
2.7 Use case: Closeness as an epidemic propagation tool

As a way to showcase the applicability of our method in real-life usage, we propose a final experiment motivated by a pragmatical use case. Let us consider a cellular offloading scenario where a content provider (CP) needs to propagate a piece of information as fast as possible inside a city while consuming as less cellular resources as possible [87]. To do so, the CP decides to send the information to a single node, which has to share the information with its direct entourage. Direct (or device-to-device) communications allow users to exchange data directly between each other without having to go through a base station, thus enabling such a scenario in a real-life setting.

The CP wants to send a single copy of the information, to the most well-connected node in order to achieve maximum information propagation within the shortest amount of time. The closeness-centrality measures the mean distance (in term of hops) of a node to the rest of the graph and thus, by construction, can be used as a tool to quantify which individual could share a piece of information most rapidly to the rest of the network.

2.7.1 Epidemic propagation

We conduct an epidemic propagation which starts at time t, for a total duration of 600 seconds. Initially, a single node is *infected* (has the information), and this node infects anyone that stays within its communication range for 5 consecutive seconds, the time we consider sufficient to transfer the information. Newly infected nodes also contaminate their neighbor



(a) 6 a.m.

(b) 8 a.m.

(c) 12 p.m.

(d) 9 p.m.

Figure 2.11: Comparison of epidemic propagation at different hours of the Luxembourg dataset. We select a source node based on its actual centrality (node), or the geo-centrality.

nodes, and so on. We proceed to two disseminations with the same topological conditions: in one case, the first infected node is chosen thanks to the closeness while in the second case, the node is chosen thanks to geo-c. We then measure the evolution of the number of nodes having received the information as a function of time.

To determine which node has the highest closeness, we simply compute the closeness centrality of all nodes at time t, given t is the beginning of the infection.

The *geo-c* closeness is computed over the window $W_t = [t, t + 600[$, with a square size of 20 meters. To select the first infected node with based on the *geo-c* closeness, we first select a square from the first quantile (i.e, a square S which satisfies $\textit{geo-c}(W_t, S) = 1$) at random,

and if nodes are found within the boundaries of square S at instant t, we randomly choose one of them as the first node to be infected. If no node exists within that specific square, we pick another one.

Contrary to the closeness-centrality propagation, our *geo-c* closeness propagation is not deterministic, since we randomly chose a square and then randomly chose a node within that same square. We run the *geo-c* propagation 100 times to accurately quantify the mean and standard deviation of the propagation over time.

2.7.2 Propagation Results

Luxembourg. We start our propagation scheme at four different times of the Luxembourg dataset, each corresponding to different traffic patterns. On Figure 2.11, we plot the results of the propagation evolution as a function of time, with the error bars for the *geo-c* representing the standard deviation around the mean number of nodes infected. Since the population density is different depending on the hour of the day, the numbers on the y-axis may differ from one time of the day to another.

On Figure 2.11a, we observe the propagation scheme at 6 a.m., which corresponds to a time where the node density slowly rises. Firstly, the number of infected nodes is nearly similar using closeness and *geo-c* closeness, with a slight advantage in favor of the geo-c. This showcases that even though the nodes selected for infection through our *geo-c* method may not be exactly the same as the one obtained through a regular closeness calculation (hence the standard deviation around the geo-c), they exhibit the same behavior in terms of propagation.

The previous observation took place during a very-low population density scenario, and we now take a look at a high-density scenario on Figure 2.11b, where the propagation takes place during rush-hour (8 a.m.).This time, the two curves are strikingly similar, almost superimposed, and once again demonstrates the ability of the *geo-c* to locate nodes with the highest closeness. It does not come to a surprise that this time of day yields very close results, since massive traffic jams reduce the mobility of high-closeness nodes, making them easily geolocalized.

We have shown in section 2.5 that anticipation during the mid-day often performs poorly. However, when we start the propagation at 12 p.m., the node closeness and mean *geo-c* closeness still provide the same results. During this period, however, the geo-c has a large spread

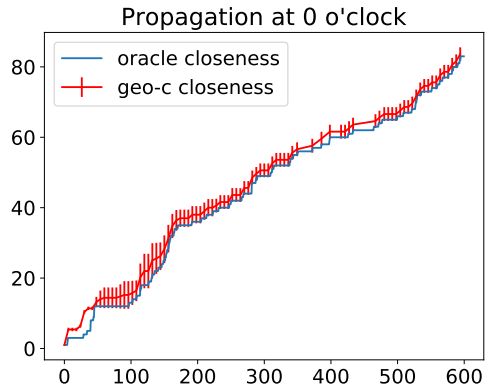


Figure 2.12: Comparison of epidemic propagation in the city of Stockholm.

around its mean, since the standard deviation reveals that the *geo-c* may grandly over or underperform the regular node closeness, during this time of day.

As the last example to showcase the similarity between the two schemes, we started a propagation at the beginning of the evening (9 p.m.). At this hour, the nodes selected through the *geo-c scheme* completely outperform the node closeness by approximately 30 percent.

While this experiment showcases the similarity between epidemic propagation based on the actual closeness and the *geo-c*'s closeness in a vehicular context, we propose to go a step further and generalize those results using a different type of mobility dataset.

STOCKHOLM. We now run the same dissemination experiment over a pedestrian dataset, the Ostermalm dataset [50], a mobility trace taking place in the city of Stockholm. The trace has a duration of approximately five hours, with a total of 2,400 nodes moving within a 5,872 m^2 area. Contrary to Luxembourg, the population does not vary significantly throughout the trace, holding a constant value around 60 nodes at any instant.

In Figure 2.12, we show the propagation over time when using the Stockholm dataset. Since the population does not vary over time, starting the propagation at any time of the dataset yield approximately the same results. We can easily observe the two very similar behaviors for the *geo-c* and node closeness for this pedestrian propagation. The standard deviation is small, showing the little variability in the results even though we ran the *geo-c* experiment 100 times. This once again shows that the *geo-c* is able to find nodes with approximately the

same closeness value, enabling equivalent propagation.

In conclusion, this experiment shows that the *geo-c*, even calculated over a large window, can find nodes with a closeness similar to the node with the highest real closeness value. If we could learn the *geo-c* of 10 minutes over the duration of a typical workday, the *geo-c* can be used as a way to approximate the closeness of nodes during upcoming workdays at the sole cost of querying the centrality map based on time of day and location of the node.

2.8 Conclusion and future work

Throughout this chapter, we explored the correlation between the centrality of nodes and their geographical position. To do so, we defined a geographical centrality of small areas (squares), based on the centrality of nodes located inside of them. We evaluated this idea over two vehicular datasets, by comparing an estimation of the current real centrality of nodes to the passed geographical centrality of the square they are in. Our evaluation showed that up to 80% (resp. 90%) of the 20% highest (resp. lowest) centrality values can be successfully estimated, based on their position. By investigating the success rate as a function of the time of day, we saw that the success rates of different centralities strongly variate according to population. Our finding is, the higher the population, the more reliable the centrality estimation can be done from the position, for the highest centrality nodes. Some centralities are less sensitive to this, notably the closeness, which yields good results regardless of population or time.

Our future work consists of turning the ideas presented here into a pragmatical solution for position-based centrality estimation. We will first study in greater detail the failures in the predictions by analyzing how far-off the prediction was from the actual centrality, to achieve a higher success rate. Then, we aim to implement a database containing the list of areas with their associated centralities according to time of day. It would require to pre-process mobility traces to calculate the centralities beforehand, but once such calculation is done by leveraging the day-to-day redundancy in traffic patterns, we can easily estimate the centrality of a user at the sole cost of querying the database, independently from the user's identity or centrality complexity.

A great deal of my work is just playing with equations and seeing what they give.

Paul Dirac

3

Characterization of D2D throughput through empirical validation

The interest of adopting device-to-device (D2D) communications is evident in many real use cases, ranging from cellular traffic offloading to localized social networks [30, 87]. Nevertheless, nothing guarantees that the quality of the direct link is sufficient to fit the application requirements. The literature, although rich in theoretical analyses and proposals of opportunistic communications, still lacks experimental work that focuses on finely understanding D2D links *in real setups*. The majority of the work either consists of simulations that often sidestep multiple parameters behind an actual D2D communication or implements a D2D framework/application but restrained to short-range laboratory settings [57].

It is surprisingly difficult to find answers to simple questions such as "*what is the expected transfer rate between two smartphones if they are 25 m apart?*" or "*how far can these two smartphones communicate reliably?*" In this chapter, we focus on characterizing the link quality of device-to-device communications based on empirical measurements. The particularity of our experimental methodology is that, besides the wireless environment, we also consider other sources of influence on the quality of the link such as the device brand and the strategy to establish direct links. Surprisingly, the quality of the link is highly dependent on the device

and software that the users carry.

As a summary, the contributions of this chapter are:

- Android D2D APIs review. We recap the currently available high-speed D2D tools in stock Android. As some APIs are proprietary, their inner-workings are not always disclosed, which requires indirect analysis to assess their behavior.
- Data collection app. To acquire data through these APIs, we designed Ocat, an application whose purpose is to implement all currently available APIs and store all collected information for post-processing.
- Measurements campaign. We detail our experimental procedure to collect data by using Ocat on Android smartphones from three different brands, considering both Google Nearby and Wi-Fi P2P APIs. We vary the distance between the devices.
- Heterogeneity's characteristics analysis. We explore the performance difference between the devices. We notably reveal that performance is not as straightforwardly correlated to the hardware or the distance as expected.
- Distance-to-throughput model. Finally, after a thorough analysis of the collected data, we propose a model to estimate the goodput-to-distance upper-bound, using the best performing devices.

This work aims to be a stepping-stone on how to accurately model D2D communications between two Android devices in a real-life setting. While there are still limited efforts to leverage D2D communications as an applicable data-exchange solution [30], we are hopeful results provided in this work entice the community to do so.

3.1 Related work

We first looked into studies where the distance between two devices is estimated from the received signal strength. This is a standard practice in wireless sensor networks [48, 54, 111], vehicular networks [23, 97], and in mobility detection of smartphones [62, 81].

Given the ubiquitous nature of Wi-Fi in modern devices, some studies evaluate the performance of the different Wi-Fi standards, which can be useful depending on the smartphone's supported standard. Zeng *et al.* explore the relationship between overlapping Wi-Fi standards (e.g., 802.11n/ac using the same frequency) and power consumption [113]. Saha *et*

al. extend this idea by focusing on the characterization of the relationship between Wi-Fi throughput of smartphones and battery consumption using several popular models [91].

Some authors have previously raised the idea of establishing the throughput based on the distance between two entities. Chowdhury *et al.* approximated the distance-to-throughput by first fixing a path loss equation and then observing the distances covered by each modulation in the case of IEEE 802.11n standard [24]. Qayyum *et al.* proposed to measure the distance-to-throughput using a mobile application in Android [84]. Although sharing some goals with us, their work was restrained to short-range D2D links as they focused on Bluetooth links. Neto *et al.* proposed to estimate the contact throughput by taking into account the speed of a mobile node [76] and a Wi-Fi access point. They transferred data and attempt to establish a relationship between speed, RSSI, and throughput.

Several authors have implemented D2D communications using Android devices Android. For instance, Keller *et al.* propose a cooperative streaming system named Microcast [57]. They divide a file into chunks, with the base station sending a chunk to one of the devices which in return has to share the chunk with its entourage and eventually receive chunks by eavesdropping on other devices. In this specific case, all devices are connected through the same access point, which is unlikely in reality. Other examples of full-fledged D2D data sharing solutions exist, but they all assume the devices communicate within a short range [11, 63]. Another issue is the assumption that devices can maintain several simultaneous D2D connections, while in reality this is not guaranteed and switching from one Wi-Fi Direct connection to the next may create a latency of several seconds [38]. On top of this, while these frameworks may be functional, they are not found in stock Android and are therefore not always available to developers nor have guaranteed support.

While all these works are substantial to their domains, they often ignore the importance of distance as a parameter for the throughput estimation. Fortunately, some authors shed light onto this issue. The closest works to our study have investigated the idea of tuning throughput according to the distance. For instance, Chowdhury *et al.* [24] observed that the Wi-Fi protocol adapted its rate according to the RSSI, and therefore proposed to calculate the amount of transferable data between a Wi-Fi access point and a mobile node according to the distance between them. Neto *et al.* [76] take this idea and empirically try to verify this model by measuring the throughput between a mobile node (car) and a base-station. Last, Qayyum *et al.* [84] take an empirical measurement of the throughput according to distance

using a mobile application.

An essential difference between our study and the previously mentioned works is the method used to estimate the throughput between two devices according to the distance separating them. Chowdhury *et al.* [24] took the theoretical link speed as specified in the Wi-Fi 4 (IEEE 802.11n) standard, and tried to fit a piece-wise and polynomial function to obtain the link speed from the RSSI. Then, they estimate the RSSI according to distance by using a log-distance propagation model. Qayyum *et al.* [84] first empirically measure the distance to throughput by implementing a Bluetooth data transfer scheme in a mobile application, and then propose a distance to throughput model using cubic spline interpolation. Compared to our model, they report a notably lower throughput, and in their case, shorter range ($\approx$10 m). As detailed in the rest of the chapter, our model takes the best of these two approaches.

3.2 Stock Android High-Speed D2D APIs

To establish the upper-bound of D2D data-rate according to the distance in modern Android devices, we decided to go for an empirical approach and thus must carry out field measurements. To this end, we first take a look at currently available D2D APIs in Android.

As of this thesis, there are two supported APIs for high-speed D2D in stock Android, namely **Wi-Fi P2P** [45] and **Google Nearby Connections** [43]. The former is an implementation of the Wi-Fi Direct standard [21], while the latter is a closed-source framework to share files, strings (e.g., URLs), or even data streams (e.g., VoIP) with surrounding devices.

Some other third-party solutions may exist for D2D communications or even hardware modifications, but we remind the reader that we intend to investigate stock Android as the majority of users do not modify their OS. In other words, the whole purpose of this study is to find a model which emulates the throughput to distance behavior of off-the-shelf smartphones.

3.2.1 Wi-Fi P2P

Wi-Fi P2P is the Android implementation of the Wi-Fi Alliance's Wi-Fi Direct standard. Peer-to-peer (P2P) devices communicate through *P2P Groups*, which are dynamically formed by electing a device as the *P2P Group Owner*, which embodies the role of the access point. As

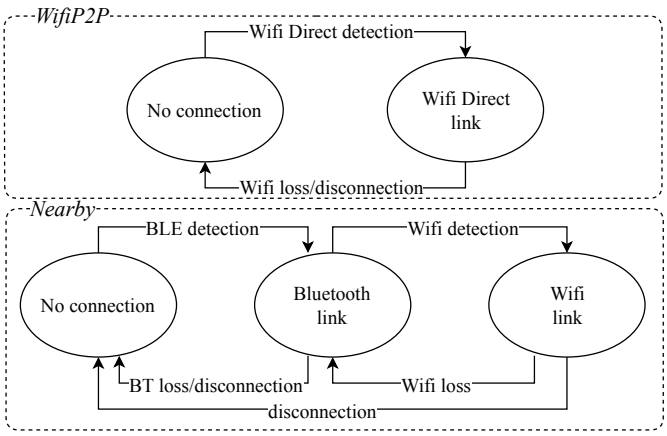


Figure 3.1: Simplified connection process of both APIs.

specified in the standard, a P2P device can concurrently act as a P2P Group Owner and as a P2P Client of another group.

Once two P2P devices have detected each other, one (or both) can initiate a Wi-Fi P2P connection, as shown in Figure 3.1. The Wi-Fi Direct standard imposes the connection to be set up through the Wi-Fi Protected Setup protocol, thus forcing the connection to be approved manually by the user(s) via a pop-up notification. A P2P Group Owner is decided, and IP addresses are then configured through DHCP. In a nutshell, this all the Wi-Fi Direct does, and thus, all this API does. After the assignment of IP addresses to both devices, a socket has to be opened on the newly available D2D interface to establish a two-way communication tunnel between the two devices.

For data to be exchanged between the two D2D devices, the responsibility of transport and application protocols choices is left to whoever will design the data exchange (e.g., OS/mobile app programmer). Our implementation of the application protocol built on top of the Wi-Fi Direct will be thoroughly detailed in Section 3.3.1.

Limitations. The standard's inner-workings have been thoroughly reviewed in the literature [21], yet, as of this thesis, the Android implementation reveals significant technical restriction. While the standard does not state any limitations in the number of clients in a P2P group, we were unable to maintain more than one D2D connection using the Wi-Fi P2P API. This issue is independent of the role embodied by the device (P2P Group Owner or P2P

Client), in fact from our experiments we found that the Wi-Fi P2P framework systematically interrupts any previously existing Wi-Fi P2P connection in favor of the new one. Surprisingly, an Android device can connect to one P2P Group and a regular 802.11 AP at the same time, hence proving multiple simultaneous Wi-Fi connections are technically possible in Android [38]. The single Wi-Fi P2P connection restriction seems to be more of a design choice rather than a real technical limitation, thus possibly mitigated in the near future.

3.2.2 Nearby

Nearby Connections was released in 2017 by Google as a high-throughput D2D framework to completely abstract network-related complexities of the connection so that developers can focus on application features rather than communications technicalities. For the rest of this thesis, we refer to Nearby Connections API as *Nearby* for simplicity's sake, but keep in mind other variants of this API exist (Nearby Messages and formerly Nearby Notifications, shut down in late 2018).

Other than the application level functions (and callbacks) available to developers, there are close to no technical specification of the inner-workings of Nearby. As such, we do not know which network or transport or application-level protocols are used when transferring data.

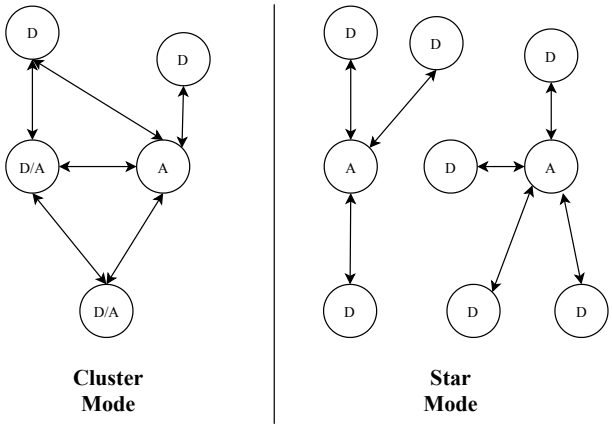


Figure 3.2: Illustration of both topologies supported in Nearby

Topological restrictions. There are two supported types of topology: cluster and star*. An illustration of the topological difference can be found in Figure 3.2. The devices have to choose a role before starting the connection process; they can be either discoverers or advertisers (noted as D, A or D/A), along with a service name. As we can observe, cluster topology acts in a completely decentralized fashion, where any device can accept/start a connection from/to any other device. On the other hand, the star topology is more restrictive, since advertisers can accept all incoming connections but cannot initiate a connection with another advertiser, and a discoverer can only be connected to a single advertiser.

Nearby leverages all the commonly available wireless technologies found in off-the-shelf smartphones, namely Bluetooth Low Energy (BLE), Bluetooth, and Wi-Fi. The topological differences between star and cluster seem to be imposed by the technology used. For instance, cluster topology uses Bluetooth for data exchange which enables connections to multiple devices simultaneously, while the star topology focuses on leveraging the speed of Wi-Fi links for high-throughput through an adviser acting as an AP. This also explains why the star topology is restrictive, since Android client devices may only be connected to a single AP at a time. In this chapter, we solely consider the star topology for this API due to significantly higher bit-rates enabled by the Wi-Fi, since we aim to establish the upper-bound of D2D reachable rates.

Using the `adb` software [42] and a spectrum analyzer, we unveil that the connection process is actually three-fold, as summarized in Figure 3.1. First, a BLE beacon is sent to notify all users within communication range that a server is available (*advertiser*) or a client is looking for a server to connect to (*discoverer*). Once two devices have detected each other, they first establish a Bluetooth connection in order to begin the data transfer as soon as possible. During this Bluetooth data transfer, the two devices attempt to establish a Wi-Fi connection; if they succeed, they automatically switch the data transfer over to the Wi-Fi link. If the Wi-Fi link drops due to poor signal, they fall back to the Bluetooth connection. The Wi-Fi standard and carrier frequency are set by the advertiser, which acts as an AP.

After the establishment of the D2D link, the file transfer can start. Nearby triggers regular callbacks to notify the user-space of internal framework events, such as a status update on the transfer of the file.

*A new mode, point-to-point, was released since this thesis was written [Google]

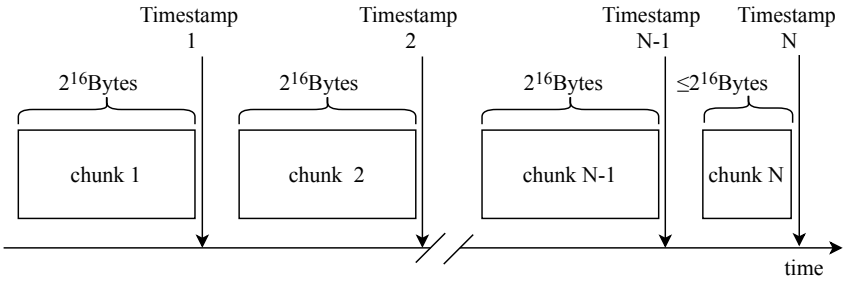


Figure 3.3: Representation of application-level data arrival in Nearby.

3.2.3 Goodput measurement

We focus on Wi-Fi P2P and Nearby because they both enable higher transfer speeds than BLE and Bluetooth. Unfortunately, there is no integrated way in neither Wi-Fi P2P nor Nearby APIs to obtain the throughput of the D2D wireless link (i.e., the data exchange rate between the two Wi-Fi interfaces). Thus, instead of obtaining the throughput, we consider the *goodput* in the remainder of this chapter. The goodput is the data rate measured at the application level.

To obtain the goodput, we have to look at the behavior of the APIs. Nearby triggers a callback every time a *chunk* is received. A chunk is $\leq 2^{16}$ bytes-long (the Nearby API sets this size). As shown in Figure 3.3, we store a timestamp at the arrival of each chunk in order to calculate the mean goodput after the file has been entirely sent. We apply the same idea in our implementation of the Wi-Fi P2P API. For both Nearby and Wi-Fi P2P APIs, we calculate the mean goodput as:

$$\text{Mean Goodput} = \frac{\sum_{n=2}^{N} \text{sizeChunk}(n)}{\text{Timestamp N} - \text{Timestamp 1}}. \tag{3.1}$$

We start at $n = 2$ because we cannot obtain the timestamp before the arrival of the first chunk.

3.2.4 Network-related data collection

To better understand D2D characteristics, we need to collect all possible accessible network-related information/statistics when using either APIs. As we have not found any straight-

forward way to obtain extra information using Nearby and Wi-Fi P2P APIs, we use a third API supported in stock Android that solves the issue at least for the Nearby case. This API, called WifiManager [46], is initially meant to handle client Wi-Fi connections in Android. We exploit the fact that once Nearby establishes a D2D link through Wi-Fi, this link acts as a standard Wi-Fi connection; thus, it becomes possible to use WifiManager to query the Wi-Fi interface to obtain extra information. Thanks to this workaround, we obtain:

- the received signal strength indication (RSSI, in dBm)
- the link speed (in Mbit/s)
- the used frequency (in MHz)

Unfortunately, this technique exclusively works on the client-side of the connection, due to the AP-side information being inaccessible in Android.

Once again, while there arguably could be complex software solutions such as installing a custom Android firmware on the device or tampering the hardware, this falls outside of the scope of this work since we focus on off-the-shelf hardware running stock Android to generalize our results easily.

3.3 Experimental procedure

To assess the upper-bound performance of D2D data exchange rates in modern Android smartphones, we conduct a measurement campaign to empirically evaluate the performance of the Nearby and Wi-Fi P2P APIs.

We designed and implemented Ocat (Opportunistic communications assessment tool), a mobile application whose purpose is to measure the connectivity characteristics between two or more devices using (so far) either Nearby or Wi-Fi P2P. In Figure 3.4, we show the user interface of the application. The process consists of generating random files of 10 MBytes, transmitting them between the devices, and storing all the gathered information in a logging file for post-process analysis. It is useful to remind that we only collect information once the data reaches the application level at the receiver side since we cannot access network-related information at the operating system level.

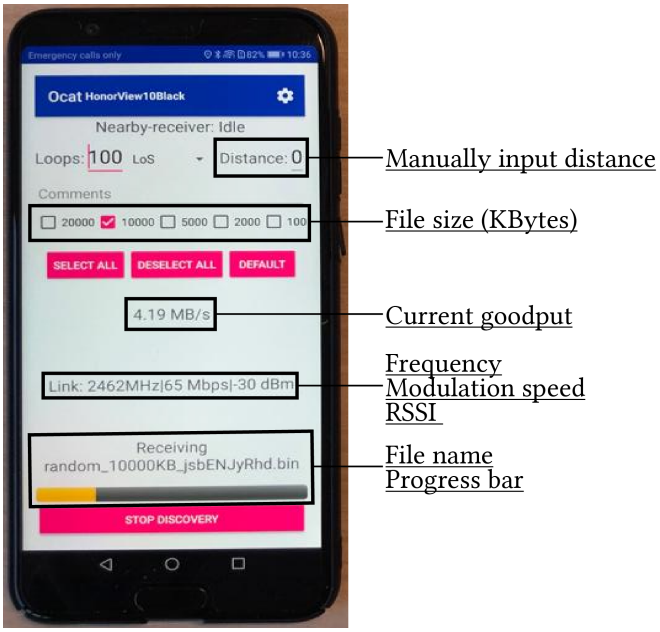


Figure 3.4: Ocat user interface.

3.3.1 Ocat

In Figure 3.5, we show the protocol exchanges implemented in Ocat. In order to have a richer view of the exchanges between the devices, we keep the data and control planes separate. The purpose of the control plane is mainly to keep the synchronization between the transmitter and the receiver although it also communicates other useful information for the log to be intelligible such as the name of the file being sent, the name of the device currently transmitting, and so on. As precisely estimating the distance between two devices is complex using nothing but the available information, we decided that the user must manually input the real distance. This distance, set on the transmitter side, will be shared through the control plane to the transmitter to simplify the post-process analysis.

Among technical issues we came across while implementing this app, we noticed that Nearby prematurely notified the transmitter of transfer completion. Using this synchronization scheme allowed us to enforce strict transmission rules so that two files are never sent concurrently, which is a requirement for a fine-grained measurement of file transfer rates.

Last but not least, once the file transmission is completed, we must ensure that the file has

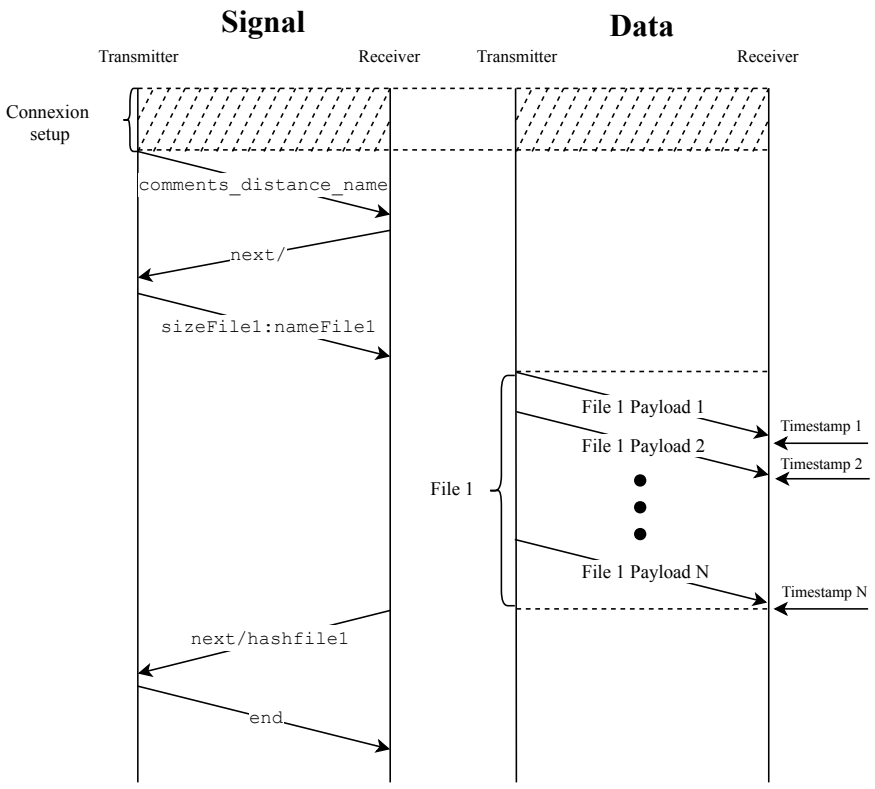


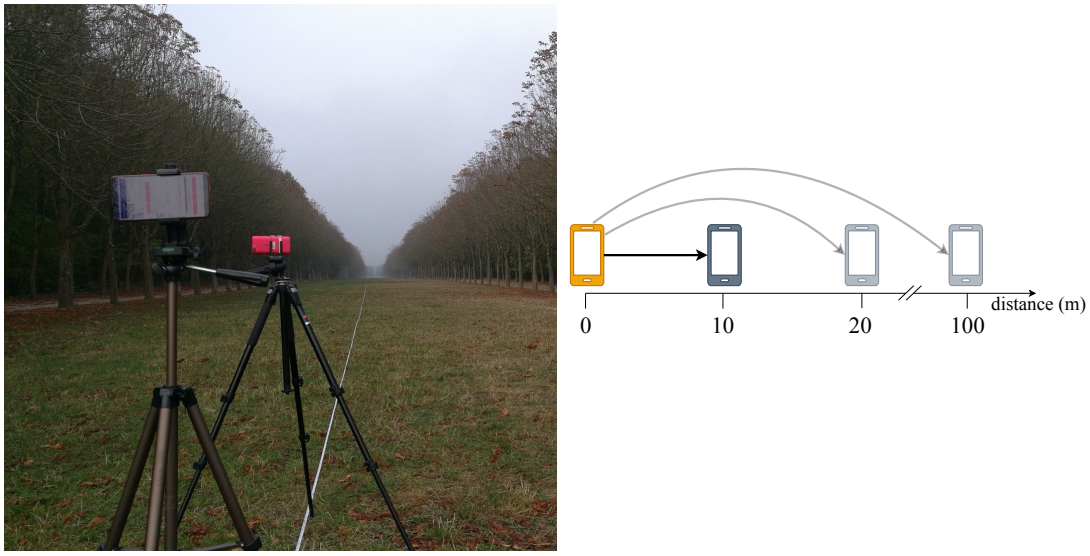
Figure 3.5: Representation of the synchronization protocol implemented in Ocat.

been sent correctly – since we cannot make any assumptions in regards to the APIs reliability. Thus, the receiver must compute the cryptographic hash of the received file and send it back to the transmitter, which in return will compare this hash and the one from the original file. In case of comparison failure, the same file will be sent several times until a maximum retry has been reached, which in our case was set to 5. Additionally, the receiver may request the next file in the transmission queue with the `next/` command.

This synchronization protocol is built on top of both APIs, for the sake of having consistent logs between both APIs, thus streamlining the post-process analysis.

3.3.2 Distance-to-Goodput methodology

We now propose the following methodology to obtain the goodput as a function of the distance. In this experiment, we laid on the ground a measuring tape which we used as a marker



(a) Photography of the experimental procedure. Both smartphones run the Ocat application.

(b) Representation of the gradual spacing of the devices.

Figure 3.6: General measurement methodology.

to know the exact distance d between the transmitter and receiver devices. We put the smartphones on tripods, both of them set at the height of 1.3 meter. We start by putting them 1 meter apart, as seen in Figure 3.6a, and then we gradually increase the distance, as represented in Figure 3.6b. Using Ocat, we connect two devices using either Wi-Fi P2P or Nearby (an option in the application). Through an active D2D connection, the transmitter sends 100 randomly generated files to the receiver. Once all files have been sent, we move the receiver to the next mark and start the next round of transmissions. We repeat the process until the devices are too far to establish a connection.

In practice, a plethora of parameters related to the wireless medium have to be considered, such as shadowing, refraction, or fading. To limit as much as we could the influence of these parameters, we conducted our experiments in a rural environment with little to no external interference (Figure 3.6a). This is compliant with our goal of measuring the maximum reachable transfer speeds. We checked for interference using a spectrum analyzer by verifying that no signal above noise-level (-100dBm in our case) was detected on the 2.4 GHz and 5 GHz bands.

3.4 Devices testing

In the experiments, we make use of six smartphones from three different brands: 2×Samsung Galaxy S8, 2×OnePlus 5T, and 2×Honor View 10. From the product specification and using a spectrum analyzer in an interference-free environment, we derived Table 3.1.

Model	#antennas	Nearby	Wi-Fi P2P
Samsung S8	4×4	2.4 GHz (Wi-Fi 4)	5.8 GHz (Wi-Fi 5)
OnePlus 5T	2×2	5.1 GHz (Wi-Fi 5)	5.1 GHz (Wi-Fi 5)
Honor View 10	1×1	2.4 GHz (Wi-Fi 4)	5.1 GHz (Wi-Fi 5)

Table 3.1: Technical characteristics of the smartphones used in the experiments.

The characteristics of the devices are, as expected, heterogeneous. We can see that, using Nearby, the Samsung and Honor devices use Wi-Fi 4 (802.11n), which has a 20MHz bandwidth, while the OnePlus device uses Wi-Fi 5 (802.11ac), which may have a bandwidth ranging from 20 to 160 MHz. All of the tested devices, when using the Wi-Fi 5 standard, automatically set their bandwidth to 80MHz. Interestingly, when using Wi-Fi P2P, all devices adopt Wi-Fi 5.

3.4.1 Close-range communication performance

To better understand the differences between the devices regarding signal transmission and reception capabilities, we carried out an experiment using Ocat by putting a pair of devices side-by-side and then measuring the signal strength on the receiver side.

In Figure 3.7, we show the mean RSSI measured between all pairwise combinations of the devices. On the *x*-axis, we show the transmitter's name, and on the *y*-axis, the receiver's name. Thus, by reading a column, we observe how a particular phone performs as a transmitter, while a row shows its performance as a receiver.

If we observe the first two columns on the left, where the Honor View 10 devices act as senders, we can underline that the mean signal strength measured by all received brands ranges from -29 to -35 dBm, thus performing significantly worse than the other brands. When observing rows 3 and 4 (in the middle), where the Samsung S8 devices act as receivers, we could conjecture that they are poor receivers to some extent, as they seem to have a signal strength ranging from -12 to -18 dBm.

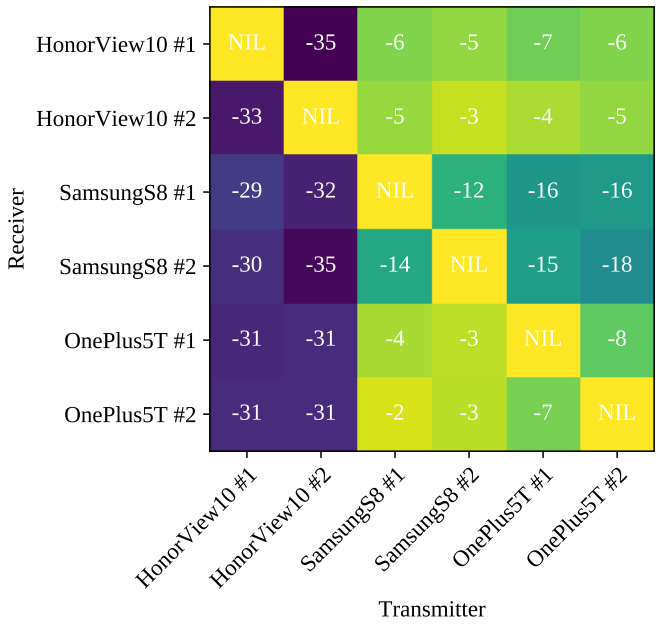


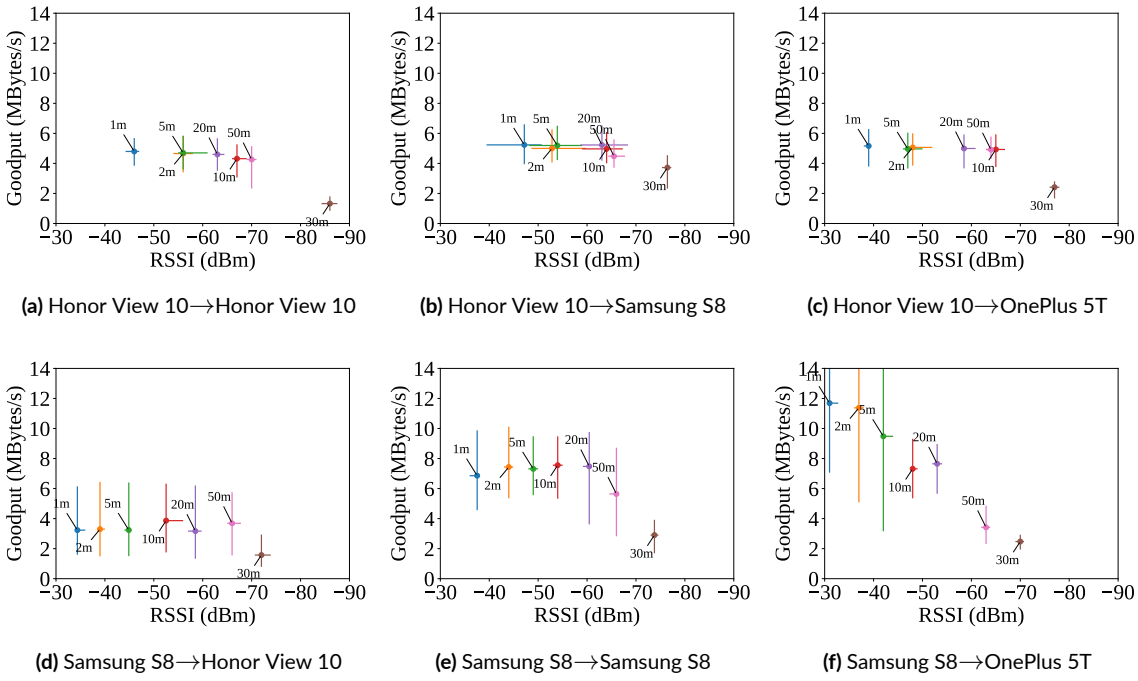
Figure 3.7: Mean reception signal heatmap for Nearby for all devices tested. For this experiment, devices were placed side-by-side.

From this preliminary experiment, we could hypothesize that D2D communications using off-the-shelf devices present device-dependent asymmetrical signal transmission or reception characteristics.

3.4.2 Medium-range performance characteristics

Verifying the performance of D2D communications when they are located precisely next to each other is arguably not enough to validate performance differences. Because of numerous parameters (e.g., shadowing, fading, multipath), the devices may tend to perform equally in a real-life situation, contrary to what the side-by-side signal reception might suggest. As an effort to point out differences in the tested hardware and to weed out devices with possibly underwhelming performances, we test all of the six smartphones with the distance-to-goodput methodology established in the previous section. We choose a stepping of 1-2-5-10-20-30-50 meters and keep longer distances for the best performing devices in a later section.

The results of this measurements campaign can be found in Figure 3.8 and Figure 3.9,



(a) Honor View 10⟶Honor View 10 **(b)** Honor View 10⟶Samsung S8 **(c)** Honor View 10⟶OnePlus 5T

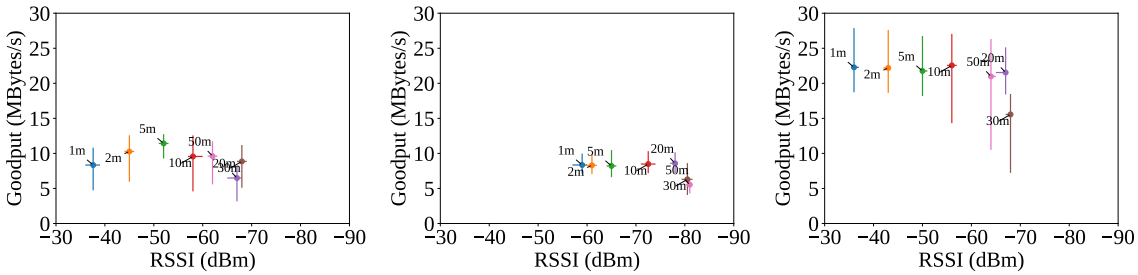
(d) Samsung S8⟶Honor View 10 **(e)** Samsung S8⟶Samsung S8 **(f)** Samsung S8⟶OnePlus 5T

Figure 3.8: Overview of medium range experiments on all using smartphones Wi-Fi 4.

with each subplot containing the result of measurements done with a particular setup, a setup being a pair of sender and receiver. Each colored dot represents a set of measurements done at a specific distance. The x-axis represents the signal strength measured, with the horizontal error-bar representing the 5th-95th percentile. The y-axis represents the median goodput measured, with the vertical error bar representing the 5th-95th percentile. The error bars may be asymmetrical.

Let us first have a look at Figure 3.8, where all the transfers are done using Wi-Fi 4. We can observe, in all of the plots, that the signal at 30 meters seem to display more mediocre performance than at other distances, surprisingly even when the devices are farther apart such as 50 meters. This observation notably showcases that we cannot monotonically correlate distance with goodput, as the 50-meter performance exhibits not only better signal reception but also higher goodput than 30 meters, regardless of being a sender or a receiver.

Focusing on the first row, where Honor View 10 devices act as senders, the goodput seems relatively similar, about 5.5 MBytes/s independently of the receiver. For the 30-meter mark,



(a) OnePlus 5T→Honor View 10 **(b)** OnePlus 5T→Samsung S8 **(c)** OnePlus 5T→OnePlus 5T

Figure 3.9: Overview of medium range experiments on all using smartphones Wi-Fi 5.

as previously observed, the goodput and received signal strength are consistently lower than the other measurements. In Figure 3.8b, where the Honor View 10 sends to the Samsung S8, we can observe that the signal strength spreads, represented by the horizontal error bars, overlap with each other. This implies that there is an extreme signal strength variance in this particular setup, thus making it hard to uniquely match a specific RSSI to a distance.

In the next row, where the Samsung S8 emits, several differences appear. In Figure 3.8d, where the Samsung S8 sends to the Honor View 10, we can observe underwhelming performance in the goodput when compared to the rest of the row. This difference is partially explained by the fact that the Honor View 10 has a single antenna, hence disabling the MIMO feature of Wi-Fi 4 while the other models have multiple antennas. Nonetheless, when we compare this figure to Figure 3.8b, where the roles of sender and receiver are reversed, the goodput medians are higher, approximately 5.5 MBytes/s compared to 3-4 MBytes/s, along with a notably larger spread of the goodput.

The above observation notably highlights that we cannot estimate the goodput of D2D communications in a pairwise manner, since sending and receiving capacities seem to be asymmetrical depending on the model used.

If we compare Figures 3.8e and 3.8f, when the OnePlus 5T acts as a receiver, the goodput is significantly higher at close range than when the Samsung is a receiver. This is particularly counter-intuitive since Samsungs have 4×4 antennas and OnePluses have 2×2, one may think that Samsungs should automatically have better goodput. Nevertheless, it is not the case, and OnePluses seem to also have better signal reception, with all points shifted to the left of the plot when compared to when the Samsung acts as a receiver.

Let us now observe the results of the D2D communications under Nearby when using Wi-Fi 5, meaning when a OnePlus acts as the sender. In Figure 3.9a, the goodput from one distance to the next is not as stable as previously observed on Wi-Fi 4. For instance, at one meter the goodput is approximately 8 MBytes/s, and at 5 meters it is 13 MBytes/s. Seeing at 10 meters it goes back down to 9 MBytes/s, we once again demonstrate that distance and goodput measurements cannot be monotonically correlated.

Figure 3.9b shows different results. First, an obvious shift to the right can be observed, thus showing that this particular setup yields poor signal reception on the Samsung. While the Samsung has four antennas and the OnePlus has two antennas (thus OnePlus to Samsung D2D communications should use 2×2 MIMO), the goodput is only $\approx$8-9MBytes/s; this is approximately as fast as the Honor who only has a single antenna. Once again, it shows that there seems to be a performance issue regards Samsungs acting as receivers since no MIMO-proportional growth of goodput is observed we could hypothesize that MIMO is either deactivated or not function properly. On the other hand, when a OnePlus sends to another OnePlus (Figure 3.9c, the goodput goes up to 23-24MBytes/s). When compared to the performance of the Honor, the growth is more than two-fold, thus showing that MIMO can function adequately with the Nearby API.

We first demonstrated that the sending and receiving capacities are, in fact, asymmetrical. In other words, when considering two different devices, the goodput between the two strongly depends on which one is currently receiving and sending. Secondly, even when a phone has MIMO capabilities, it may not function when using Nearby, and therefore does not always correctly work while using D2D communications. Thirdly, when considering the same sender at the same distance, receiving devices may have a very different signal strength readings.

As this study aims to establish an upper-bound of D2D data exchange rates, in the remainder of this chapter, we explore results for the OnePlus 5T smartphones acting both as sender and receiver. The reasoning behind this choice is threefold: the OnePluses have the best goodput performance and overall good signal strength measurement regardless of the sender. Secondly, neither the Samsung S8 nor the Honor View 10 uses the same Wi-Fi standard for both Wi-Fi P2P and Nearby (see Section 3.1) – this is likely to produce biased results which are difficult to interpret. Thirdly, since we are using the same device model as sender and receiver, we can ignore the issue of the asymmetrical characteristics.

3.5 RSSI for goodput estimation

As previously explained, we can only access the RSSI on the receiver side, as Android does not give access to the transmission power of the sender. The RSSI, in dBm, is calculated as:

$$\text{RSSI}(d) = P_t + G_t + G_r - L - PL(d), \tag{3.2}$$

where d is the distance between the transmitter and the receiver, P_t is the transmission power, G_t is the transmitter's antenna gain, G_r is the receiver's antenna gain, L is the loss of the system, and $PL(d)$ is the path loss according to distance d. For the off-the-shelf smartphones that we use in our experiments, most of these parameters are not disclosed.

We first assume that smartphones transmit at constant power, meaning that P_t is fixed. We simplify the equation by letting $A = P_t + G_t + G_r - L$. This is done since we can only measure the RSSI, hence making it impossible to differentiate the terms. We have then:

$$\text{RSSI}(d) = A - PL(d). \tag{3.3}$$

3.5.1 Modeling the RSSI

We present in Figure 3.10 the RSSI measurements between two OnePlus 5T following the measurement methodology presented in Section 3.3. We chose the following stepping: from 1 to 10 meters we use a 1 m stepping, from 10 m to 100 m we use a 5 m stepping, and from 100 m we use a 10 m stepping until the signal is lost. While at short distances (1 m to 20 m), the strength of the signal seems to be monotonically decreasing, it is not the case throughout the entire experiment. For instance, from 30 m to 50 m, there is an apparent increase in the signal strength even though we put more distance between the devices.

To accurately model this path loss behavior, we consider three different models, namely the free-space path loss propagation model [14], the log-distance path loss model [14], and the accurate version of the two-ray ground-reflection model (as opposed to the approximation often found in the literature) [86, 97].

Free-space model The free-space model, sometimes known as the Friis transmission equation, is the expected signal reception in an optimal environment. Its path-loss compo-

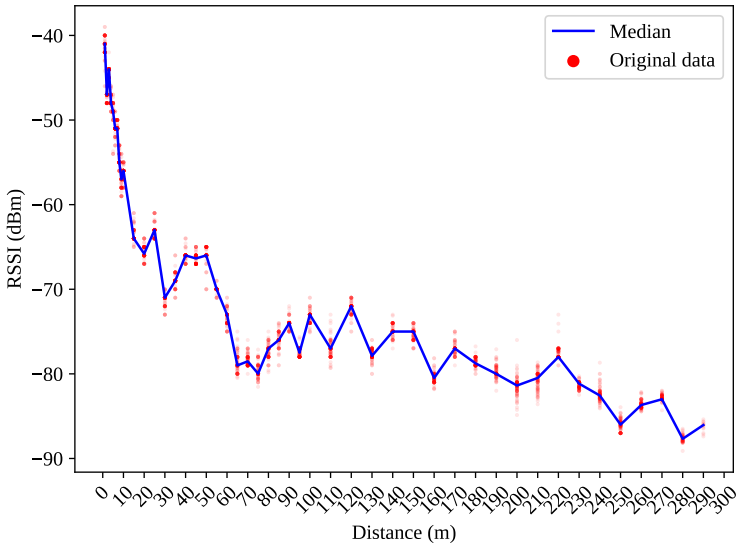


Figure 3.10: RSSI to distance measurements using the OnePlus 5T devices.

nent can be written as:

$$PL_{\text{freespace}}(d) = -10log_{10}\left[\frac{\lambda^2}{(4\pi)^2 d^2}\right] \tag{3.4}$$

with λ the wavelength in meters, and d the distance in meters between the transmitter and receiver.

Log-Distance model Other more general models exist, for instance, the Log-Distance path loss model which can be tuned to fit any environment by setting an attenuation exponent. For instance, the indoor environment of a building will have a greater attenuation exponent than an outdoor field. It is formulated as [14]:

$$PL_{\text{logdistance}}(d) = PL_{\text{freespace}}(d_0) + \eta 10log_{10}\left[\frac{d}{d_0}\right] \tag{3.5}$$

with d_0 a close-in reference distance (typically 1 meter), d the distance in meters, and η the attenuation exponent. When the attenuation exponent is equal to 2, this formula is the same as the free-space, therefore η has to be equal or superior to 2.

Two-ray ground-reflection model The idea behind this model is that the receiver obtains two copies of the same signal, the original one and a copy reflected from the ground. The principle is to calculate the phase difference between the two copies of the signal to verify whether the interference is constructive or destructive, based on the heights of the transmitter and the receiver. In our case, these heights are equal because of the tripods, which guarantees that the line of sight distance between the two devices is $d_{los} = d$. The distance of the reflected signal is $d_{reflect} = \sqrt{d^2 + (h_r + h_t)^2}$, where h_r and h_t are the heights of the transmitter and receiver devices, respectively (again, equal in our setup). The phase difference as $\phi = 2\pi \frac{d_{los} - d_{reflect}}{\lambda}$.

To obtain the final model, we also need the reflection coefficient, which gives the capacity of the ground to reflect an electromagnetic wave. It is given by:

$$\Gamma_{\perp} = \frac{\sin\theta - \sqrt{\varepsilon - \cos^2\theta}}{\sin\theta + \sqrt{\varepsilon - \cos^2\theta}} \tag{3.6}$$

where θ is the angle of incidence of the reflected signal based on the heights of the transmitter and receiver, and ε is a fixed parameter based on the material.

The two-ray ground reflection model is then written as [97]:

$$PL_{\text{tworay}}(d) = 20\log_{10}\left[4\pi\frac{d}{\lambda}|1 + \Gamma_{\perp}e^{i\phi}|^{-1}\right]. \tag{3.7}$$

Using Euler's formula, we replace the complex exponential in the equation as:

$$PL_{\text{tworay}}(d) = 20\log_{10}\left[4\pi\frac{d}{\lambda}|1 + \Gamma_{\perp}\cos\phi + \Gamma_{\perp} i\sin\phi|^{-1}\right]. \tag{3.8}$$

As the modulus of a complex number yields a real number, we use this final equation:

$$PL_{\text{tworay}}(d) = 20\log_{10}\left[4\pi\frac{d}{\lambda}\sqrt{(1 + \Gamma_{\perp}\cos\phi)^2 + \Gamma_{\perp}^2\sin^2\phi}^{-1}\right]. \tag{3.9}$$

Fitting models onto experimental data We estimate the parameters for each model through a best-fit approach. As we only have the RSSI at each receiver, we need to estimate the global gain of the system A (see Equation 3.3) for all models. To find which path loss

model leads to the best fit, we perform a least-square curve fitting using *lmfit* [77]. We remind the reader that, depending on the *PL* model used, different parameters are used to find the best fit. These parameters are listed in Table 3.2, where η is the attenuation exponent of the log-distance model and ε is a parameter that depends on the material of the reflection surface and h_r/h_t the height of the receiver/transmitter.

Table 3.2: Summary of the models and their parameters.

Path-Loss Model	Param 1	Param 2	Param 3
Free-space	A	-	-
Log-Distance	A	η	-
Two-Ray ground-reflection	A	ε	h_r/h_t

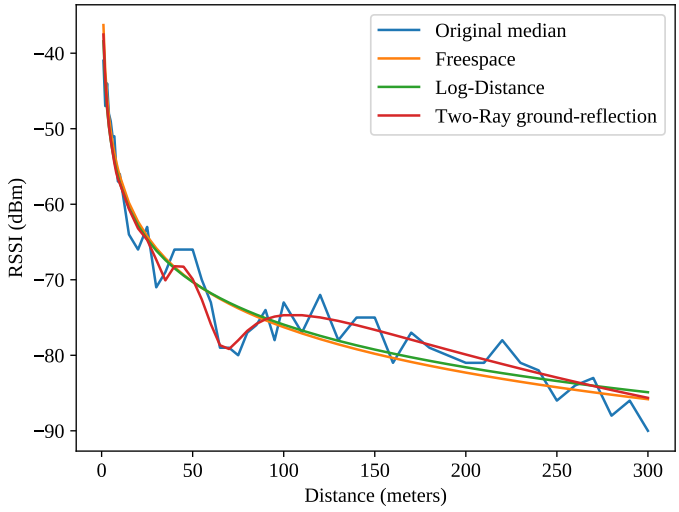


Figure 3.11: RSSI Models after least-square fitting. The two-ray ground ground-reflection model is the only model capturing the signal increase while augmenting the distance.

We show in Figure 3.11 how the different models fit our measures. The free-space model leads to a pretty good fitting by using a global gain $A \approx 10.46$. The log-distance model seems to exhibit the same behavior, however, the best-fit yields a gain of $A \approx 8.3$ and the attenuation exponent is fitted to $\eta \approx 1.8$; the problem is that η is under the minimum value of 2 which corresponds to the propagation of a signal in free-space. Thus, we consider this

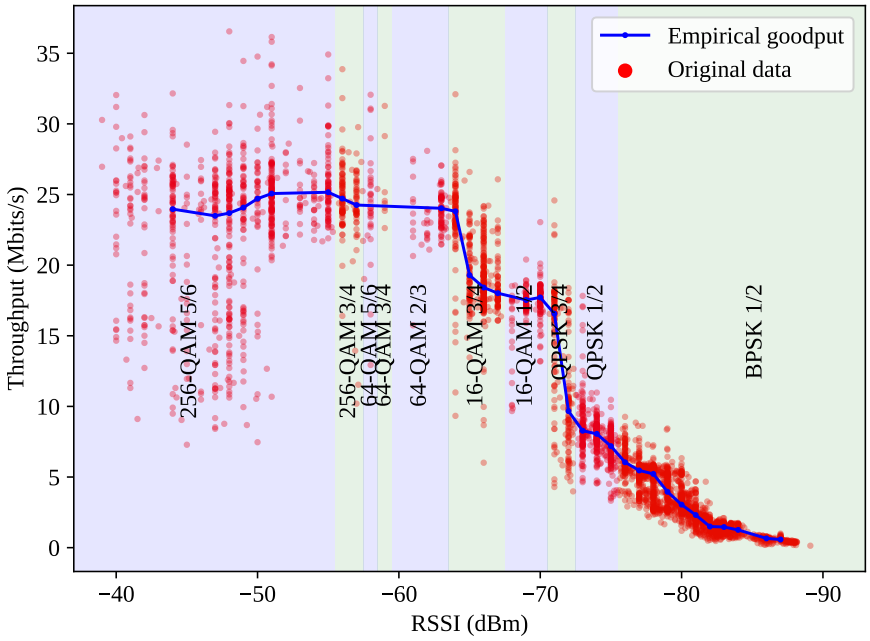


Figure 3.12: Empirical RSSI-to-goodput relationship with the indication of the modulation used.

fit to be invalid and ignore it since fixing this parameter to a minimum of 2 is equivalent to calculating the free-space model.

The two-ray ground-reflection model gives the best results. It can capture specific phenomena such as the signal increase observed between 30 m and 50 m (which is the consequence of constructive interference due to the ground reflection). The algorithm fixes the parameters $A \approx 9.19$ dB and $\varepsilon = 1.009$. It is not a surprise that the algorithm gave $h_r = h_s \approx 1.38m$, which corresponds to the height of the phones on top of the tripods.

3.5.2 RSSI and Goodput

In order to derive a distance-to-goodput model, we need to establish if a clear correlation between a given RSSI value and a goodput exists. We know from the Wi-Fi 5 standard [25] (a.k.a. 802.11ac, used by the OnePlus 5T devices) that, for each value of RSSI, a particular modulation is chosen in order to achieve the best trade-off between throughput and error resilience. Typically, modulations reaching the highest throughputs are quite sensitive to noise

Table 3.3: Android Wi-Fi 5 theoretical maximum throughput using 80MHz bandwidth & 2*2 MIMO, 400ns GI

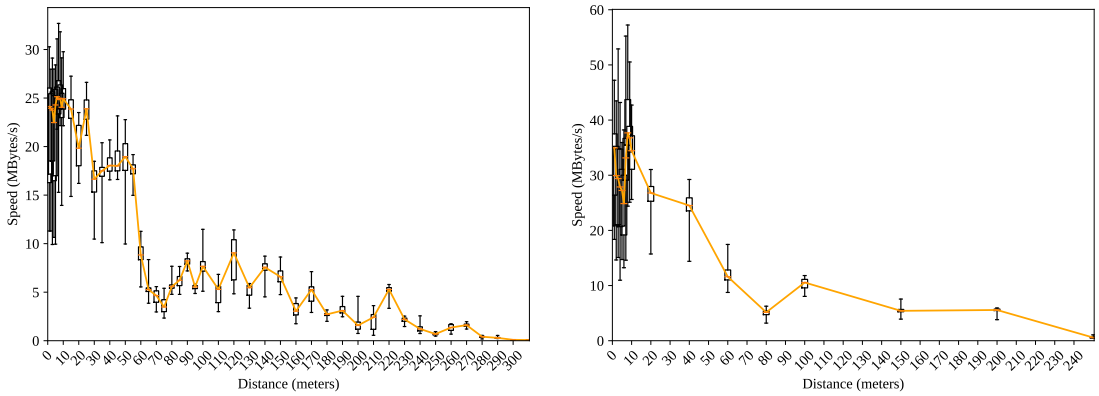
MCS	Data Rate	Modulation	Redundancy	Max RSSI	Min RSSI
9	866	256-QAM	5/6	-	-55
8	780	256-QAM	3/4	-56	-57
7	650	64-QAM	5/6	-58	-58
6	585	64-QAM	3/4	-59	-59
5	520	64-QAM	2/3	-60	-63
4	390	16-QAM	3/4	-64	-67
3	260	16-QAM	1/2	-68	-70
2	195	QPSK	3/4	-71	-72
1	130	QPSK	1/2	-73	-75
0	65	BPSK	1/2	-76	-

and are unsuitable for long-distance communications.

Recall that, in Android, the modulation used by the Wi-Fi interface is not disclosed; still, it can be inferred from the link speed if the number of MIMO antennas is known. We compiled Table 3.3 from the Wi-Fi 5 standard to better understand the behavior of the throughput. The BPSK modulation does not have a minimum RSSI as it continues to operate until the signal is lost, and the same logic applies to the 256-QAM, which operates no matter how high the RSSI is.

The units we consider are dBm for the RSSI and MBytes/s for the goodput. In Section 3.2.3, we calculated a mean goodput per file sent, and we now apply the same idea to the RSSI. The RSSI given by WifiManager is not guaranteed to be up-to-date for each chunk. We mitigate this issue by weighting the RSSI of each chunk using the transmission time of the chunk to establish the mean RSSI of the file.

As the choice of a specific RSSI value is not left to the user, we cannot guarantee that all possible values of RSSI are covered. However, by iteratively moving the tripods back and forth several times, we were able to cover a wide range of possible values, from -39 dBm to -90 dBm. In Figure 3.12, we present the results of the RSSI-to-goodput value. Each blue dot in the figure represents a sample (a file sent by Ocat) associated with a goodput, and we superimposed the RSSI range of all modulations to give a physical layer perspective to the reader. The red line represents the median values of all values of RSSI, given that at least 50 samples over the same RSSI are available.



(a) Nearby **(b)** Wi-Fi P2P

Figure 3.13: Goodput measurements from OnePlus 5T to OnePlus 5T connections with both APIs. The box-plots' whiskers include 95% of the data.

We observe, on the left part of the plot, where the RSSI values are covered under the 256-QAM modulation, a broader spread of the goodput around the median when compared to other modulations. For instance, between -40 dBm and -50 dBm, the goodput varies from 7 MBytes/s to 35 MBytes/s with a median around 24.5 MBytes/s. Note that up until the usage of the 64-QAM modulation, approximately the same median value is found, except that the spread around the median is significantly smaller, ranging from 20 MBytes/s to 30 MBytes/s.

We observe that the median does not significantly change in the range -40 dBm to -62 dBm despite five different modulations. Since prior works have found a clear correlation between modulation used and energy consumption in smartphones [113], this reveals a potential energy consumption issue. As the goodput using the modulations between 256-QAM 5/6 and 64-QAM 2/3 are unequivocally equivalent, it seems that using the 64-QAM 2/3 modulation from -40 dBm to -62 dBm could be a better alternative as we know that it consumes less energy. Also, there are two significant drops in the measured median goodput, at precisely -63 dBm and -72 dBm. While it would seem intuitive to think that they correspond to a modulation change, they are not correlated to a particular modulation.

Strictly speaking, the upper-bound should consist of all maximal values measured at each

RSSI. However, we notice on Figure 3.12 that there are outliers that may strongly bias our model with values close to non-reproductible. As a way to propose a bound reasonably close to the upper-limit we will consider the median as it adequately corresponds to the close-to-optimal RSSI-to-throughput D2D relationship we want to approximate.

The median of Figure 3.12 can be easily approximated by a linear piecewise function $\widehat{\text{goodput}}(x)$, where x is the RSSI:

$$\widehat{\text{goodput}}(x) = \begin{cases} 24.26 & x \geq -64, \\ -0.36x + 42.35 & -64 < x \leq -71, \\ -0.79x + 66.44 & -71 < x \leq -82, \\ -0.24x + 21.13 & -82 < x. \end{cases} \tag{3.10}$$

Now that we have the goodput as a function of the RSSI, we need to extend it to become a function of the distance. This is the topic of the next section.

3.6 Goodput estimation according to distance

3.6.1 Analysis of empirical observations

Let us first note that we were able to maintain a D2D connection up to 280 m distance using Wi-Fi P2P and up to 310 m using Nearby. This difference is explained by the fact that Wi-Fi P2P is solely based on Wi-Fi 5 (5 GHz band) while Nearby may fall back to Bluetooth (2.4 GHz band), as explained in Section 3.2.2. Nonetheless, when Nearby remains at the 5 GHz band, the distance limit is around 280 m, exactly like Wi-Fi P2P. To the best of our knowledge, there is no reference in the literature reporting D2D experiments that achieve such long distances using off-the-shelf smartphones using Android.

In Figure 3.13, we show the goodput results obtained from two OnePlus 5T devices communicating using both Nearby and Wi-Fi P2P. When we observe the results for Nearby (Figure 3.13a), several patterns emerge. On short distances (1 m to 20 m), the throughput seems relatively constant, ranging from 23 MBytes/s to 25 MBytes/s. The dispersion around the median is however quite large, with values as high as 33 MBytes/s and as low as 10 MBytes/s – but this is consistent with our observations in the RSSI analysis in Section 3.5.2, where a close-range communication generated more spread around the median.

As for the Wi-Fi P2P API (Figure 3.13b), we see that the large variance pattern is also found within short distances in the range of 1 m to 20 m. The median ranges from 35 MBytes/s to 39 MBytes/s, which is significantly higher than the speed reached with Nearby. We believe that this difference in short distances is due to a software bottleneck. Another hint is the previously studied RSSI-to-goodput relationship (Section 3.5.2), which showed no increase in throughput, albeit using modulations with higher theoretical speeds. Hence, we can deduce that Nearby's maximum goodput is, in fact, software-related more than hardware-related.

Going back to Figure 3.13a, we observe an increase in the goodput in the range 30 m to 50 m. We were expecting this behavior as it corresponds to the constructive interference due to ground reflection (as discussed in Section 3.5.2). This enables the devices to maintain a high goodput, ranging from 13 MBytes/s to 17 MBytes/s; notably, such a high goodput is achieved within medium-range distance.

The next pattern can be seen in the range 60 m to approximately 270 m. We observe a sawtooth behavior. This is a consequence of the low RSSI over long-range communications, where any obstacle may increase or decrease the signal (e.g., leaves on the ground and nearby trees). Regardless, the goodput ranges from 1 MByte/s to 7 MBytes/s, which is still a significant value.

Over even longer distances, between 280 m and 310 m, the goodput measurements are less accurate due to Nearby's behavior. When the Wi-Fi connection drops between two devices, which can happen when the RSSI is around -90 dBm, Nearby automatically falls back to the Bluetooth connection. Nearby still allows devices to exchange data, but this time with a significantly lower bit-rate between 10 KBytes/s to 100 KBytes/s.

3.6.2 Model of the goodput as a function of the distance

Having explored the multiple aspects affecting the D2D communications in our experimental setting, we now propose a final model to estimate the goodput based on the distance between two Android devices.

We use the $PL_{\text{tworay}}(d)$ model to estimate the path loss and fix the inner parameters from the least-square estimation, where $A \approx 9.19$ (see Section 3.5.1). From Equation 3.3, we derive:

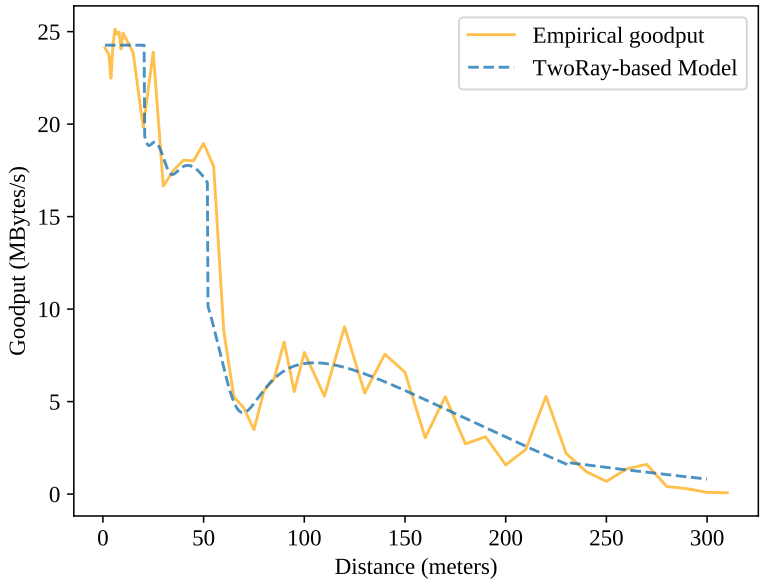


Figure 3.14: Comparison between the empirical data and our model for Nearby.

$$\widehat{\text{RSSI}}(d) = 9.19 - PL_{\text{tworay}}(d). \tag{3.11}$$

To obtain the goodput in function of the distance, we combine Equations 3.10 and 3.3:

$$\text{goodput}(d) = \widehat{\text{goodput}}(\widehat{\text{RSSI}}(d)). \tag{3.12}$$

In Figure 3.14, we show how this model fits the dataset. The original data is the same as the one presented in Figure 3.13a, but in Figure 3.14 we only show the median for the sake of readability. Our model presents the expected characteristics that we observed in practice, notably the sudden surge in goodput in the range 30 m to 50 m along with the significant drop after 70 m.

The proposed model, which is the upper-bound of D2D communication in off-the-shelf Android smartphones, allows the research community to estimate the quantity of exchangeable data based on the distance between two devices.

3.7 Conclusion and future work

In this chapter, we explored the current state of D2D communications in stock Android devices. To this end, we first designed and implemented Ocat, an Android application that eases the measurement process of D2D communications through both Google Nearby and Wi-Fi P2P. As a result, we were able to establish a model for the upper-bound goodput of D2D communications as a function of the distance between the communicating devices.

We first confirmed the fact that smartphones have asymmetrical signal reception performance [69], and we notably highlight the fact that goodput performance is also asymmetrical. In other words, accurately quantifying the D2D link from both sender and receiver side, not pairwise, is critical to model D2D interactions accurately.

We showed that the Nearby API can maintain a goodput of 25 MBytes/s in the 1 m to 20 m range. Within the same range, Wi-Fi P2P can reach up to 39 MBytes/s, showing that Nearby possibly lacks software optimization to achieve similar speeds. In our findings, we note that the ground reflection has a significant impact on goodput performance, which can be positive within medium distances between 30 m to 50 m, enabling a steady goodput rate between 13 MBytes/s to 17 MBytes/s. Surprisingly, devices were able to communicate at relatively long distances (280 m to 310 m).

We also revealed that D2D communications in Android smartphones potentially have an energy consumption issue: due to the software maximum of Nearby, the goodput using a 256-QAM modulation and 64-QAM modulation is the same. Since using 256-QAM consumes more energy, deactivating it for D2D-based Wi-Fi communications is recommended on current smartphones using Nearby.

In our future work, to better assess D2D data sharing capabilities, we intend to exchange data between several devices at the same time and different distances. Ultimately, this work enables a new sort of characterization for D2D communications by establishing how much data can be exchanged between devices through a spatiotemporal contact. We believe that our work will significantly help researchers and application designers understand the possibilities and limits of D2D communications using Android devices.

Errors using inadequate data are much less than those using no data at all.

Charles Babbage

4

Computing realistic and adaptive capacity of D2D contacts

Designing and implementing an opportunistic network remains a challenge [5, 55], most notably because of the highly dynamic nature of the topology. Individual nodes have their own mobility patterns, and communication links only exist when two nodes get in communication range of each other. An accurate characterization of such interactions is then a must-have step.

The literature is full of outstanding contributions that assess the underlying phenomena governing opportunistic device-to-device communications [50, 84]. In particular, we have today a good understanding of inter-contact patterns in such networks [80]. Unfortunately, there are still critical open questions regarding the *contact* characterization. While inter-contacts depend primarily on the mobility of the nodes, characterizing a contact in detail also depends on other parameters such as the device-to-device communication technology. In this chapter, we investigate issues explicitly related to the contact between two mobile nodes.

A common limitation of existing works is that they assume either that a contact has infinite capacity, allowing the transfer of any amount of data during a single encounter, or

that the communication throughput during the contact follows the nominal values of the technology (e.g., Bluetooth or Wi-Fi). This leads to a sort of *fixed-rate* characterization of contacts, where either nodes are in contact (with a unique characteristic), or they are not. In this chapter, we advocate that to achieve a better characterization of opportunistic contacts, we should adopt a finer representation of the relative mobility of nodes, in order to capture the communication possibilities better.

The central premise of our work centers around the upper-bound we established in the previous chapter. We witnessed that because of the inherent characteristics of the wireless medium, the quality of the link varies with the distance between the nodes. This variation depends on the communication technology under consideration, but regardless of this, the throughput always fluctuates as a function of the distance between the transmitter and the receiver.

In this chapter, we help advance state of the art in several regards. We investigate the *impact of the different relative distance between nodes on the capacity of opportunistic links*. By capacity, we mean the total amount of data that can be transferred during the entire contact. Our ultimate goal is to establish a basis for more accurate design and evaluation of device-to-device communication strategies. We adopt an empirical methodology to achieve a characterization of device-to-device links. Our experiments involve Android smartphones equipped with the Google Nearby API, as per previously studied. We apply the observed characterization parameters to two mobility datasets, namely a vehicular one in the city of Luxembourg and a pedestrian one in Stockholm. We confront our observations with the traditional fixed-rate contact characterization strategy and make some observations. We confirm the fact that considering only the duration of a contact and a fixed throughput is far from enough to capture the actual capacity of the contact.

Although the quantitative analysis that we showcase is dependent on the communication technology, as well as the type of devices, we considered during the data collection campaign mentioned in the previous chapter. It also depends on the environmental conditions. The qualitative observations made in this chapter can be extended to any setup, given that the wireless technology adopts a rate-adaptive strategy. We finally propose a transformation tool that takes mobility traces as inputs and generates plausible contact capacity traces as outputs. As a summary, our contributions are:

- **Adaptive contact characterization.** We thoroughly characterize the contact capacity,

meaning the amount of exchangeable data through a contact, when we consider a fixed-rate contact and an adaptive-rate contact. We show they are not equivalent, mostly due to many contacts happening at a near-maximum range.

- **Fixed parameter selection.** Even though we advocate against fixed-rate contacts as a way to estimate contact capacity, we also propose a recommended fixed-rate contact value, to yield more realistic results.
- **Dataset breeding tool.** We propose a transformation tool that takes mobility datasets as input and generates contact datasets that include plausible capacity information for each pairwise D2D link between nodes.

4.1 Related work

In the literature, fixing a data exchange rate for a contact has been an on-going issue for more than a decade. While it obviously depends on the wireless technology used, as well as the mobility scenario and the environment, most often these issues are entirely overlooked.

Delay tolerant networks (DTN), or more specifically DTN routing/diffusion algorithms, is a field where contacts and inter-contact times are used as a way to quantify exchange opportunities [10, 20, 32, 68]. Specific algorithms focus on the fact that nodes (i.e., transmitting or receiving devices) have a finite amount of memory that limits their receiving/forwarding capabilities and forces exceptional buffer management [10, 20]. Yet, while it is a common practice to take into account hardware limitation due to memory, we are surprised to see that transfer speed is simply ignored. In other words, it is often assumed that the packet transfer between two nodes always succeeds if two nodes are within communication range (though, these algorithms assume small packet size).

Let us pick the example of the Opportunistic Network Environment simulator (The ONE), a state-of-the-art simulator aiming to simplify the evaluation of DTN algorithms in a realistic fashion [58]. For the throughput of mobile devices, they chose a conservative and simplistic estimate by setting a fixed-rate throughput for contacts. For instance, they assume that nodes using Bluetooth can exchange data at a 2 Mbps throughput within a 10 m radius and that nodes using Wi-Fi have a 4.5 Mbps throughput within a 30 m radius [58]. A recent

example of such a practice is proposed by Zhu *et al.* [118], where a contact happens with any vehicle inside a 200 m range with an exchange rate of 20 Mbps. Our study shows that such a characterization is not very precise and that distance between communicating nodes should be taken into account in order to accurately model these interactions.

As mentioned in the previous chapter, several works have proposed to estimate the throughput according to distance [24, 76, 84], utilizing this method as a way to quantify adaptive contacts. Contrary to these approaches, in this chapter we consider several new aspects. First, instead of assuming only one of the two nodes is mobile, which is unrealistic in a pedestrian or vehicular environment, we suppose that both nodes in a contact are mobile. Second, while other studies simulate random positions of nodes or real traces with a small number of nodes, we use realistic traces with hundreds, even thousands, of mobile nodes to understand how the adaptive rate affects the contact capacity in a large network.

4.2 Definitions and problem formulation

We use the consensual definition of *contacts* in an opportunistic network, which is the period of time during which two nodes have a valid wireless link and can exchange data. We note this duration as τ. Nodes are mobile, and a contact starts as soon as the two nodes are within communication range of each other. Here, we assume that all nodes have the same antenna characteristics leading to symmetric links. We also assume omnidirectional antennas. We note the distance between two nodes A and B as $d_{AB}(t)$.

In this chapter, we focus on the *contact capacity* C_{AB}, defined as the maximum amount of data that can be transferred between A and B during a contact. The contact capacity depends on the duration of the contact and the data rate of the opportunistic link between the two nodes.

In the literature, many authors simplify the problem by considering that nodes can exchange data at a *fixed rate* δ^{fixed} when within communication range. In this case, the contact capacity is simple to obtain:

$$C_{AB}^{\text{fixed}} = \delta^{\text{fixed}} \times \tau. \tag{4.1}$$

Although the fixed-rate contact characterization is simple to manipulate, it falls short in

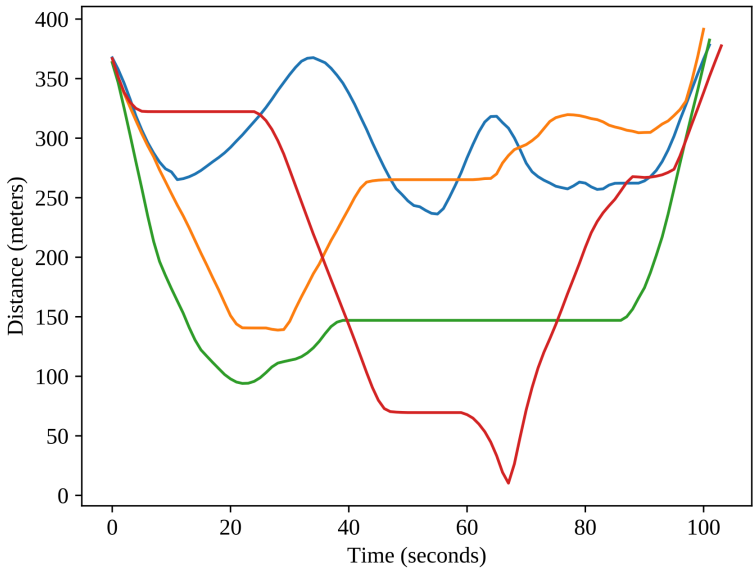


Figure 4.1: Four contacts of ∼100 seconds each. Note that the distance between the nodes show the most variable patterns.

capturing the actual characteristics of a contact in a real setup. Because of several physical phenomena, the throughput that we can get in a wireless link depends on many factors, including the distance separating the sender from the receiver and the propagation conditions. The consequence is that the contact capacity is seldom a linear function of the duration.

Although the wireless medium shows a continuous decreasing behavior in terms of signal delivery, existing technologies adopt a step-wise transmission data rate calculation as a function of the received signal strength indication (RSSI). In the previous chapter, we show this dependence in the case of IEEE 802.11ac (Wi-Fi 5), the one we consider in our experiments. As we can see, the ratio between the maximum and minimum achievable throughputs is higher than one order of magnitude. A contact between two nodes is likely to traverse several of these data rate levels; depending on the mobility pattern, the resulting contact capacity can be anything between $\tau \times \delta_L$ and $\tau \times \delta_H$, where τ is the contact duration and δ_L (resp. δ_H) is the lowest (resp. the highest) data rate authorized by the wireless technology. For these reasons, *considering only the contact duration may not be enough to characterize and analyze opportunistic networks*.

The rest of the work consists of investigating how much impact we may get from a finer

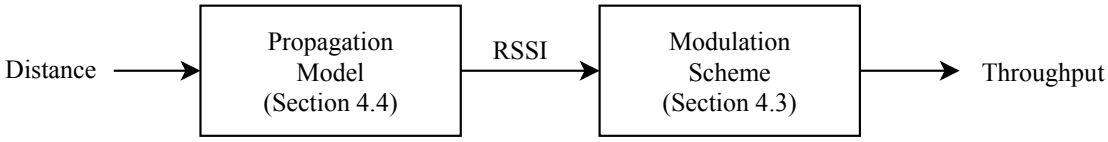


Figure 4.2: Distance to throughput model previously established. We will explore the influence of each model in the annotated sections.

characterization of contacts. The first step is to make sure that nodes do observe varying distances with regard to neighbors for the same contact duration – this is exactly what we observe in practice. In Figure 4.1, we plot four different contacts of approximately 100 seconds in a real-world pedestrian scenario. As we can see, the four patterns show that, for the same given contact duration, the distances between the two nodes may vary unpredictably from one contact to another. We can find a case (red curve) where the nodes get very close (about 10 m) before moving away again. However, we can also find a case (blue curve) where the nodes remain very far from each other during all the duration of the contact and never approach less than 250 m.

Problem statement. The estimation of the throughput according to the distance involves essentially two steps, as shown in Figure 4.2. The first one relates distance to RSSI based on some signal propagation model. The second step transforms RSSI into throughput, and this depends mainly on the modulation scheme. While there are common practices to determine the propagation model to fit a given environment, finding which modulation scheme reflects best the system is arguably more complex. To the best of our knowledge, there is no well-defined method to transform a given RSSI into a throughput. The main challenge will be to find models that can transform distance to throughput so that existing mobility datasets can also be extended to include information on the amount of data that can be transferred whenever two nodes meet.

4.2.1 Adaptive contact capacity

In opposition to the traditional fixed-rate contact characterization, whose contact capacity is given in Equation 4.1, we propose *adaptive contact characterization* as an alternative to better capture the behavior of opportunistic links.

We involve the two components of the system shown in Figure 4.2: propagation and

modulation. Let $\text{RSSI}(d_{AB}(t))$ be the RSSI measured on an opportunistic link when the nodes are at a distance $d_{AB}(t)$ (varying distance over time) and $\vartheta^{\text{adaptive}}(\text{RSSI}(d_{AB}(t)))$ be the throughput.

The contact capacity for the adaptive contact characterization is given by:

$$C_{AB}^{\text{adaptive}} = \int_{T}^{T+\tau} \vartheta^{\text{adaptive}}(\text{RSSI}(d_{AB}(t)))dt. \tag{4.2}$$

where $[T, T+\tau]$ is the contact interval.

4.3 Empirical reference link characterization

To establish a realistic reference basis, we run an experiment to observe how the relationship between RSSI and throughput behaves in off-the-shelf devices.

4.3.1 Experimental RSSI to throughput estimation

From the previous chapter, recall that we carried out day-long measurement campaigns in order to collect throughput values for different RSSIs in a near interference-free environment. For the communicating devices, we used two OnePlus 5T smartphones equipped with the Android 8.1 operating system and 2×2 MIMO antennas. We used the Google Nearby framework to establish device-to-device links between the nodes [43]. Google Nearby is a proprietary device-to-device library which uses Wi-Fi 5 (5 GHz band) for high-speed throughput.

Theoretical versus empirical throughput. One may believe that it suffices to use the technology's specification (e.g., the RSSI to MCS relationship shown in the previous chapter) to obtain realistic contact capacity estimation according to distance. To verify the validity or not of this assertion, we show the results of our experiment in Figure 4.3. The black step-wise function represents Wi-Fi 5's theoretical data rate, and red dots show the empirical throughput we observed during our experiments. The median, shown as a blue line in the plot, represents the expected empirical throughput for a given RSSI. We refer to this median as the "empirical goodput" since it represents the observed transfer capacity of off-the-shelf devices (OnePlus 5T devices in this case).

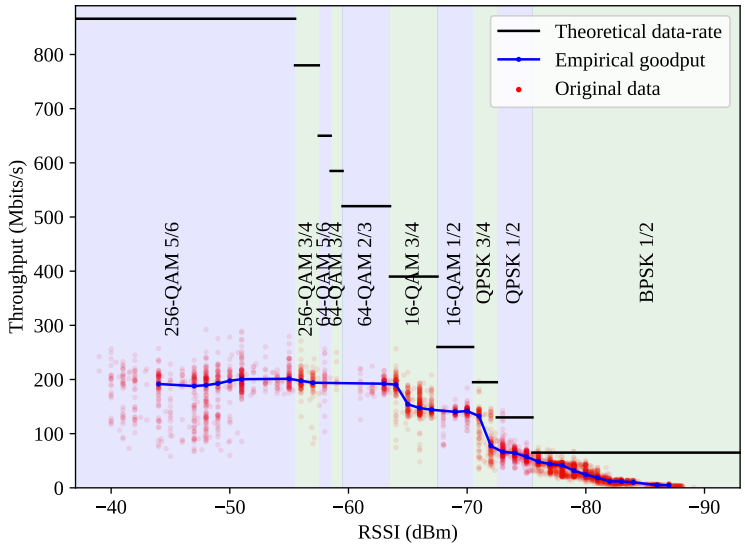


Figure 4.3: RSSI-to-throughput experimental results. The step-wise black line represents the theoretical maximum values provided by the standard.

We can make two significant observations. Firstly, the theoretical data rate is too optimistic, especially for the highest RSSI values, due to hardware limitations as well as implementation choices in the Nearby framework. Secondly, for the BPSK modulation, corresponding to the lowest RSSI levels (below -75 dBm), the experimental curve goes down to 0 Mbps at -87 dBm. As we will see later on, the fact that the theoretical curve remains horizontal at this very last modulation mode has a significant impact on the behavior of the model.

Lastly, in wireless systems, modulation schemes determine which physical modulation to be used according to one or several parameters. In the previous chapter, we took the liberty to refer as modulation scheme the mapping between a given received signal strength and an expected throughput. For clarification purposes, we remind the reader that in our measurements, as seen in Figure 4.3 as the "empirical goodput", other parameters than modulation affect the throughput. Typically, OS scheduling, buffer management, signaling, and possible QoS may interfere. In this chapter, for consistency and simplicity's sake we still refer to this relationship as the modulation scheme, since we do not have the means to differentiate the two, as seen in the following paragraphs.

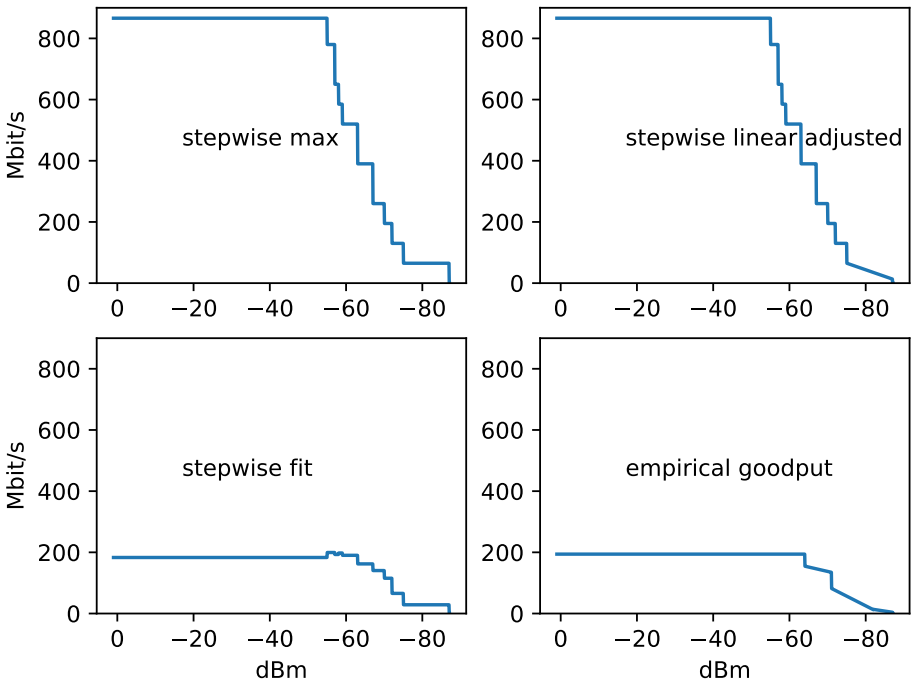


Figure 4.4: Modulation schemes considered in this work. They all have a null throughput under -87 dBm, being the minimum sensitivity of our receiver.

4.3.2 Finding an appropriate modulation scheme

In order to assess the effects of the modulation scheme, we investigate the probability density function of contact capacity for the four different modulation schemes illustrated in Figure 4.4.

- **Step-wise maximum.** This is the theoretical step-wise function taken from the Wi-Fi 5 specification. It is the simplest modulation, since it does not require experimental testing for verification.
- **Step-wise linear adjusted.** Almost the same as the step-wise maximum, except that the slowest modulation (BPSK) has its data-rate linearly decreasing until it reaches 0 at -87 dBm. This only requires knowledge of connectivity loss, thus minimal experimentation.

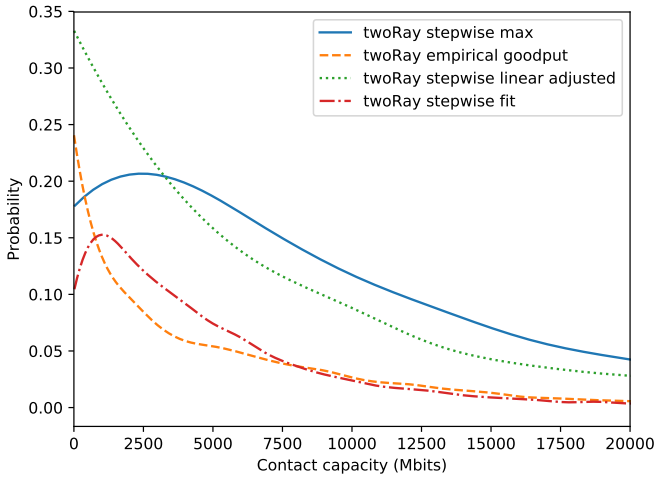


Figure 4.5: Contact capacity probability density function for all modulations using a two-ray propagation model in Stockholm.

- **Step-wise fit.** In this scheme, we move all of the steps from the Wi-Fi 5 specifications to the same levels as the experimental data. This requires heavy experimentation to correctly estimate the position of each step.
- **Empirical goodput.** For a given RSSI, the empirical goodput is the median throughput of collected samples over the RSSI (median blue curve in Figure 4.3). Similar to the step-wise fit, this modulation scheme requires heavy experimentation to be correctly estimated.

To evaluate the influence of these modulation schemes on the capacity of a contact, we first need a mobility scenario. In this section, we consider the Ostermalm dataset [50], a pedestrian mobility trace in the city of Stockholm. The trace was generated with Legion Studio [15], a mobility simulator meant for architects and designers to test out pedestrian flows in large infrastructures. The trace has a duration of approximately five hours, with a total of 2,400 nodes moving within a 5,872 m^2 area. The dataset covers a period of 5 hours and shows coordinate updates every 0.6 seconds. This high frequency of position updates is necessary as we need an accurate estimation of the distance between the nodes. The population does not vary significantly throughout the trace, holding a constant value around 60 nodes.

We also need a propagation model to transform distances into RSSI (recall Figure 4.2 and Equation 4.2). We consider the two-ray propagation model as it is the one that best fits our experimental scenario (we will discuss about the propagation models in Section 4.4.1). The results of the contact capacity PDF are shown in Figure 4.5.

Two patterns emerge. While the step-wise maximum and the step-wise linear curves show a humped behavior, the step-wise linear adjusted is the only one that succeeds to capture the behavior of the empirical goodput. Even though the step-wise linear adjusted overestimates the overall capacity of the system when compared to the empirical goodput, the most important observation here is the shape of the curve, as in a real-world scenario, we expect to observe lower capacity due to interference from neighboring communications.

The main takeaway from this experimental evaluation is the apparent necessity to take the effect of the modulation scheme into account when estimating the adaptive contact capacity. Most importantly, one has to model this carefully, to take into account the asymptotic behavior of throughput according to RSSI and, by extension, throughput according to distance.

4.3.3 Explaining the step-wise linear adjusted scheme

As we could see in the previous subsection, the step-wise linear adjusted scheme was the only approach which can capture the behavior of the experimental results, albeit a small difference with regard to step-wise maximum. Our intuition is that most contacts happen at the edge of the communication range, thus using the lowest throughput modulation (in the case of Wi-Fi 5, BPSK 1/2).

We check which of short-range or long-range contacts contribute the most, by evaluating the contact capacity of a mobility trace. To this end, we consider the two-ray ground model for the propagation. For the modulation schemes, we first assess the suitability of the step-wise maximum and the empirical goodput. In Figure 4.6, we show the CCDF of the total capacity of the network as a function of the distance between nodes. As we can see in this figure, regardless of the idealistic (step-wise max) or realistic (empirical goodput) modulation scheme, between 68% to 61% of the total capacity of the network happens due to communications occurring at more than 60 m, which shows the importance of long-distance communications and thus of BPSK modulation. As a result, the modulation scheme used over long distances should not be neglected. The use of a realistic modulation scheme, where the

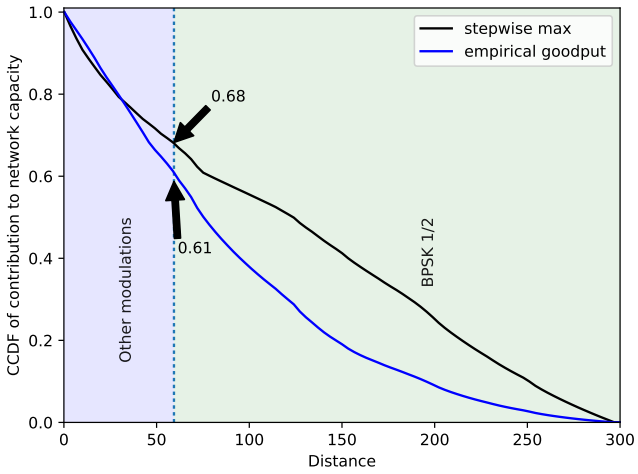


Figure 4.6: CCDF of the total capacity of the Stockholm dataset according to the distance between nodes. For the adaptive contact capacity, we here choose a two-ray propagation model and plot the capacity using two modulation schemes. Regardless of the modulation scheme, the majority of the network capacity falls under BPSK 1/2.

throughput decreases with distance until reaching zero, is recommended.

Part of the explanation also comes from the fact that, in the case of free space and two-ray ground reflection models, the area "covered" by the BPSK modulation is much bigger than those covered by the other modulation schemes. In Figure 4.7, we illustrate the differences between the areas. The bigger the area, the more neighbors are likely to communicate through longer links. Other propagation models (especially those adapted for indoors) would lead to different observations; for this reason, we will have further insights into this problem in Section 4.4.1.

4.4 Fixed vs. adaptive contact characterization

Let us now compare the capacity of contacts when they are modeled by the fixed-rate and adaptive approaches. For all contacts, we compute the capacity by applying Equations 4.1 and 4.2, respectively. To showcase the generalization of the upcoming observations, we consider two datasets, the aforementioned pedestrian Ostermalm dataset (presented in Section 4.3.2) and the Luxembourg dataset. Luxembourg LuST [28] is a vehicular mobility trace created

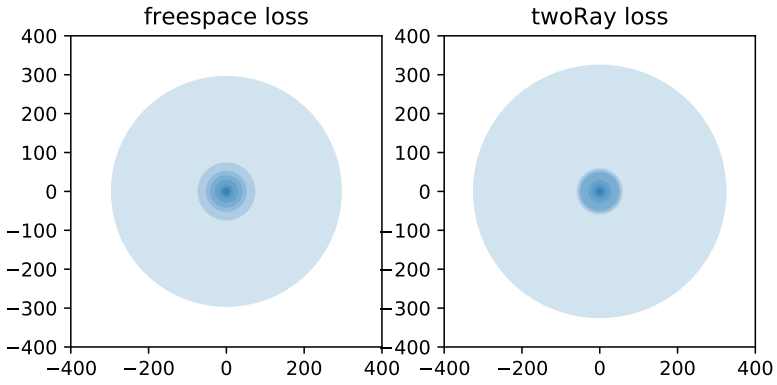


Figure 4.7: Propagation and modulation on the space. The outer disk represents the coverage zone of BPSK 1/2.

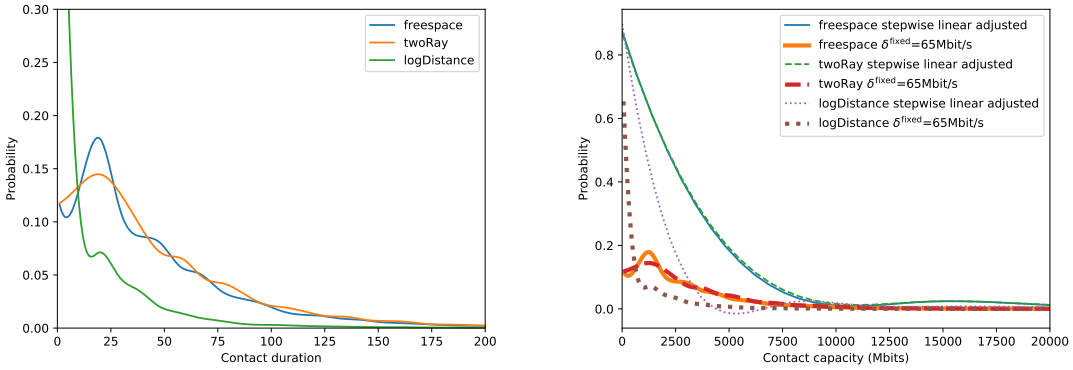
using SUMO [13], a state-of-the-art micro-mobility simulator. The simulation runs over 24 hours and covers a 20 km^2 area and a total of 167,000 nodes. The trace simulates a realistic daily traffic pattern, with peaks of population density during the morning and evening with approximately 2,500 simultaneous nodes.

4.4.1 Influence of propagation model

We previously investigated the impact of modulation schemes on the contact capacity, and we now propose to generalize our observations using different propagation models. We consider three propagation models: freespace [14], two-ray ground [97], and log-distance [14].

As previously explained, the minimum sensitivity of our devices is -87 dBm. As the propagation model determines the maximum range of communication, we consider contact duration along with contact capacity to justify the distributions. For the sake of reference, in our work, the maximum communication range using free space and two-ray propagation is $\approx$ 300 m, and for the log-distance $\approx$ 45 m (we use an attenuation exponent η of 3). We set the modulation as the step-wise linear adjusted for the rest of this section. Lastly, we also plot the contact capacity for a fixed-rate contact $\eth^{\text{fixed}} = 65$ Mbps (this throughput being the same data rate as the BPSK 1/2 for Wi-Fi 5).

For the adaptive contact, we could consider the empirical goodput since it must inherently follow a realistic contact capacity distribution. However, we know from Section 4.3.2 that the stepwise linear adjusted follows the same contact capacity distribution. As it requires



(a) Contact duration distribution.

(b) Contact capacity distribution.

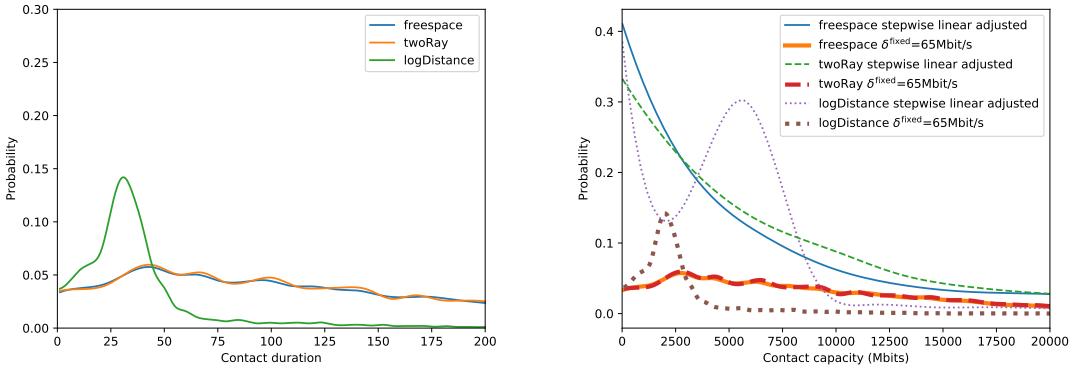
Figure 4.8: Contact duration and contact capacity distribution on a vehicular dataset (Luxembourg).

less information (i.e., does not require a full-fledged measurement campaign) than the empirical goodput, we argue that it can be used as our adaptive rate reference since we only care about the caracterisation induced, not the absolute values.

Let us first consider the vehicular case, shown in Figure 4.8. The contact duration distribution, as seen in Figure 4.8a, exhibits two different behaviors. The free space and two-ray propagation models behave similarly, with a mean contact duration around 30 seconds, whereas the log-distance model leads to a higher probability for contact durations closer to zero. This is a consequence of few contacts happening at a short distance (small coverage surface); even when short-distance contacts happen, the velocity of vehicles prevent the contact from lasting longer than a few seconds.

When observing the contact capacity distribution (Figure 4.8b), as expected, the fixed-rate curves reflect pretty well the distribution of contact duration of Figure 4.8a. What is eye-catching in this figure is the behavior of the adaptive capacity when using the log-distance propagation model. Contrarily to free space and two-ray propagation models, log-distance is much more restrictive and intended to reflect indoor scenarios. This means that the coverage zone of BPSK 1/2 with regard to the other modulations is much smaller. In this way, the ratio of neighbors communicating through low-throughput links decreases significantly.

In the pedestrian case (Figure 4.9), contact duration distribution shows a more spread



(a) Contact duration distribution.

(b) Contact capacity distribution.

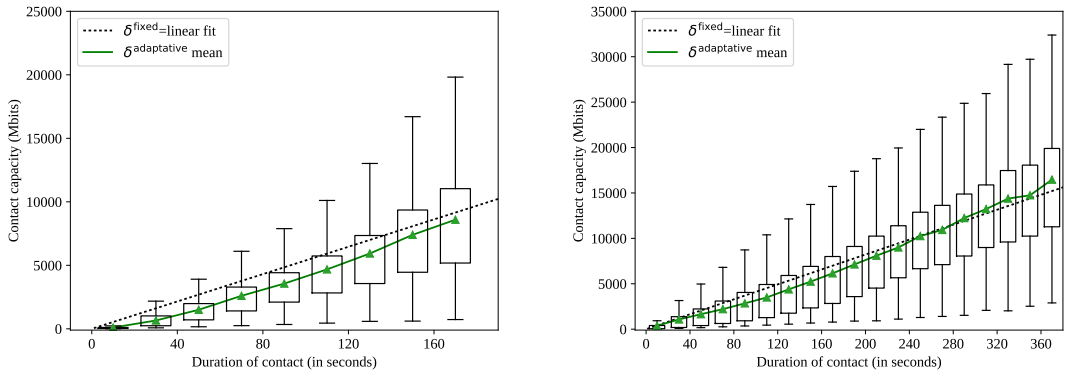
Figure 4.9: Contact duration and contact capacity distribution on a pedestrian dataset (Stockholm).

behavior than Luxembourg's. This is due to the shorter relative speeds between human beings compared with the vehicular case, which leads to longer contact duration. As such, this yields a more spread out distribution for the free space and two-ray models and a completely different shape for the log-distance model. In fact, for the log-distance model, the highest contact duration probability is found between 25 to 30 seconds.

The distribution of contact capacities in the Stockholm scenario, depicted in Figure 4.9b, has several similarities with the plots of the Luxembourg case – the fixed-rate plots follow the same shape as the distribution of contact duration, while the adaptive plot looks like that of a decreasing exponential, except for the log-distance.

The peculiar shape the log-distance with a step-wise linear adjusted modulation scheme contact capacity distribution is explained by carefully observing the behavior of nodes, which act as pedestrians. Indeed, due to the much shorter communication range in the case of log-distance, there are only two sorts of contacts. The contact is either very short, thus yielding poor capacity, or the contact is long with little distance between the two nodes. These two sorts of behaviors are a direct consequence of the pedestrian mobility behavior with short communication range; either pedestrian cross each other for a brief period, or two pedestrian walk next to each other for a longer duration thus yielding higher capacity.

We also confirm once again that the fixed-rate approach exhibits an unrealistic behavior,



(a) Luxembourg **(b)** Stockholm

Figure 4.10: Contact capacity as a function of contact duration.

and we conclude from our results that designers willing to set up an opportunistic network based on device-to-device communications should account for the type of modulation the underlying technology relies on. The influence of the environment, represented by the propagation model, is limited compared to the modulation scheme. By adopting this strategy, they may be able to obtain much more accurate results compared to what they would obtain with the fixed-rate characterization.

4.4.2 Picking a better fixed-rate value

If for some reason, one still prefers to use a fixed-rate characterization, she ought to better select δ^{fixed}. To select a better fixed-rate contact throughput value, we fit a simple linear equation on the $\delta^{adaptative}$ mean values for both Luxembourg and Stockholm scenarios, as shown in Figure 4.10. We remind the reader that in the case of contact capacity as a function of contact duration, a fixed-rate contact follows the linear equation $\delta^{\text{fixed}} = ax + b = ax$, since b is always equal to 0; a contact of 0 second obviously yielding 0 contact capacity.

The boxplots in Figure 4.10 represent the distribution of contact capacity for a given contact duration, considering an adaptive contact. With each box having a height corresponding to 50% of the data, and the bottom (resp. top) whiskers are corresponding to the 5th (resp. 95th) percentile. The curve with triangle symbols holds the arithmetic mean of all values

found inside the bin.

When focusing on the vehicular case Figure 4.10a, we can see that the linear fit for the fixed-rate contact does not seem like a good match. For contacts of less than 40 seconds long, the spread around the empirical means is quite small, but the contact capacity growth between one bin to the next is less than linear. This shows that most contacts under 40 seconds happen on the edge of the node communication range, where the data rates are the slowest. Knowing from the distribution of contact times in section 4.4 that the majority of contacts in Luxembourg happen under 40 seconds, these edge-communications concern most contacts. The slope of the δ^{fixed}-linear fit, which is the closest value that we could recommend if a fixed-rate had to be chosen, would be 53.84 Mbits/s. We advise to be wary of the use of fixed-contact, even when using this recommended value, seeing the potential incorrect characterisation noted previously.

For the Stockholm case, as represented on Figure 4.10b, the linear fitted linear function seems to be a better match. The surface in which nodes evolve inside the Stockholm dataset is much inferior to the Luxembourg one, making the average contact about 80 seconds long. Again, the slope of the δ^{fixed}-linear fit for Stockholm is 41.10 Mbits/s, which is the closest value we could recommend.

Overall, having a fixed-rate contact is more realistic in the pedestrian case than the vehicular case. We, however, remind the reader that the vast spread around the means shows fixed-rate contacts are far from being the optimal way to portray realistic contact capacity.

4.5 Contact network capacity computation tool

As a contribution to the community, we propose an open-source software to quantify the contact capacity of a mobility trace.* The software, implemented as a Python library, proposes several preset models for the propagation law and modulation scheme to meet the requirements of the user. As an incentive for users to fine-tune the software to their specific needs, we provide access to lower-level functions in case one wishes to implement customized models. The tool can handle both mobility and contact traces. A brief user-guide is provided as an appendix of this thesis.

*https://github.com/Bertier/OpportunistiKapacity

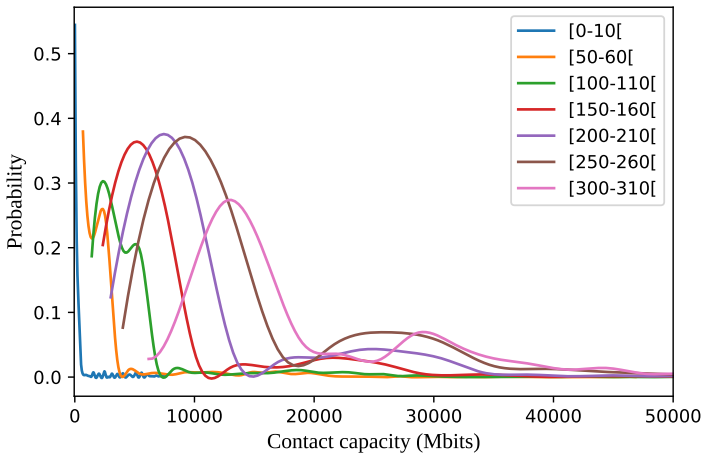


Figure 4.11: Contact capacity probability according to contact duration in the Stockholm scenario. Here, the two-ray propagation and step-wise linear adjusted modulation scheme is used.

4.5.1 Mobility traces

For this type of trace, the computation is straightforward. The software takes a mobility trace as input (i.e., a sequence of GPS coordinates) and calculates the distance between nodes at each snapshot in order to determine if there is a contact or not. A contact exists if the distance between the nodes is below the threshold set by the model. If a contact is found, it computes the capacity using the throughput estimation based on the distance between the two nodes, until they lose connectivity.

4.5.2 Duration-based contact capacity estimation

Contact traces offer the duration of each contact but not the geographical coordinates of the nodes. Thus, we cannot handle these traces as we do with mobility traces. Recall that, in Section , we presented Figure 4.1 that clearly showed that contacts of the same duration are likely to hide different mobility patterns. To circumvent this issue, we propose to rely on probability density functions that relate duration to capacity.

Let us illustrate this idea with an example. Let us assume the user chooses the two-ray propagation model along with the step-wise linear adjusted modulation scheme. Firstly, we

calculate the contact capacity for each contact duration, as seen in Figure 4.11. This figure comes from the measured data we showed in Section 4.3, and we expect to extend our reference dataset with other measurement campaigns in the future. The software computes then a contact capacity based on the duration of the contact directly from the appropriate probability density function.

As advocated in this chapter, and overall in throughout the thesis, fixed-rate contacts having a very unrealistic contact capacity distribution compared to our adaptive models, this solution should better approach the real contact capacity with significantly more accuracy.

4.6 Conclusion

In this chapter, we investigated contact characterization based on the throughput between the nodes, namely fixed and adaptive. Our adaptive contact characterization relies on the principle that one should take the distance between two nodes into account in order to accurately estimate the throughput between them. The adaptive contact characterization adopts a two-step calculation strategy: firstly, the distance is turned into a received signal strength (propagation model), then the signal strength is turned into a throughput (modulation scheme).

We applied our strategy in two mobility datasets. We observed that the essential factor when capturing the behavior of a contact is that the modulation scheme's throughput must decrease with distance until reaching zero. Also, although distant contacts have the poorest data-rate, they contribute the most in terms of global network capacity regardless of optimistic or realistic modulation schemes. By comparing the fixed and adaptive contact capacity distributions, we noticed that their distribution shapes are entirely different, thus showcasing that fixed-rate contacts are not enough to adequately capture the essence of realistic contact capacities.

We additionally provide software that computes the contact capacity of a mobility trace, either based on the distance between the nodes or, if the latter is not available, on the contact duration with greater accuracy than fixed-rate contacts. With this tool as a means to simplify the contact capacity computation, we hope that the community will get further insights into how the varying contact throughputs impact protocols and algorithms for D2D networks.

In our future work, we intend to provide a mathematical framework to calculate the contact capacity based on the duration of contact, also taking in consideration the asymmetric

features of D2D communications, thus resulting in different capacities depending on which role is embodied by each device.

Additionally, we also intend to conduct more experimental campaigns so that our tool encompasses more scenarios, typically see what the results would look like in an interference-heavy environment such as indoors.

5

Conclusion and perspectives

Throughout this thesis, we explored different aspects of device-to-device communications. We investigated D2D from theoretical, practical, and empirical points of view. In this chapter, we summarize our contributions along with our most notable results and conclude with the perspectives and questions left open.

5.1 Summary and takeaways

In this thesis, we helped advance the state-of-the-art in several different manners. From complex networks to wireless modeling from real smartphone measurements, we adopted a spread spectrum approach as a way to observe the device-to-device topic from a wide-angle. Through location-based centrality estimations, our work incentifies telecom companies to bypass heavy-load calculations. Additionally, our work notably solidifies the pragmatical aspect of D2D networks by accurately quantifying links and providing a tool for the community to measure our approach onto their own datasets.

In Chapter 2, we took an interest in the behavior of D2D contacts at the macroscopic level, approaching them in the form of contact graphs. More accurately, we investigated the relationship between the topological and geographical location of a node within a contact graph. Using our methodology, we were able to have a centrality estimate for an area, pro-

vided there was enough stability in the estimate. Doing so, we showcased that the closeness, and to some extent, the (ego)-betweenness were easily predicted from their position. Interestingly, nodes with the highest and lowest centralities have the most successful geographical estimations, revealing that these values of particular interest are rarely highly mobile.

When attempting to correlate the geographical position with a specific time of day, which is a requirement in our methodology, we observed a substantial variance in the percentage of success depending on the hour of the day. Typically, during high-population density hours such as the morning rush-hour (8-9 a.m.) yields the highest success results regardless of centrality observed. When it comes to night or mid-day density, the closeness is still able to achieve high success for the nodes with the highest values.

We found that there is a trade-off between the size of the surface (in our case, squares) and overall predictability of the nodes inside of it. Tiny squares (<10*10 m) does not yield good results because not enough nodes go through the surface to accurately predict its centrality estimate. On the opposite, a square too large (>200*200 m) also obviously yields poor results because of the natural proportionality of low-centrality nodes in contact graphs. We recommend selecting squares based on the mean percentage of predictable nodes. This way, we abstract the actual success or failure rates. When using a square size of ≈100 m results in the highest predictability rate (75%).

Takeaway 1

Estimating the centrality based on the geographical position is feasible, though highly dependant on the type of centrality, with closeness yielding the best results. The estimation success rate is highly correlated with the time of day, and the size of the predictable area.

After observing D2D relationships contact graphs, in chapter 3, we decided to establish the upper-bound of D2D links. We first reviewed the Wi-Fi P2P and Nearby Connections APIs (implemented in our data collection Android app Ocat). Using our methodology to obtain the goodput based on the distance, we obtained the upper-bound of these D2D communications, which may reach up to 25 (Nearby) and 35 MBytes/s (Wi-Fi P2P) at close distance. Even at 100 m both APIs can reach up to 5 to 10 MBytes/s and can maintain a connection up to ≈280 m.

We noticed that the goodput and the distance were not monotonically correlated. When we looked into the behavior of the received signal strength, we found that the ground reflection of the signal had a significant impact on the RSSI. When then observed that and that RSSI and goodput could be monotonically correlated. As such, we proposed to model the goodput by first estimating the signal strength, then match a signal strength to a goodput.

Not only do devices have heterogeneous characteristics (e.g., different CPU, wireless chips, MIMO antennas), through our experiments, we also revealed they have asymmetrical network performances. This means that if we consider devices A and B, both with different characteristics, the goodput of A receiving from B and B receiving from A may be different. Another interesting observation is that it is possible that higher MCS does not yield higher goodput; thus creating an energy consumption issue. We purposely chose devices with high performance and symmetrical characteristics for our model.

Takeaway 2

Modern Android devices can communicate up to nearly 300 meters distance in an outdoor setting. We can estimate the goodput between two devices using an estimation of the received signal strength transformed into an estimated goodput. The direction which the data takes in a D2D link affects the data-rate of the link, meaning D2D links are asymmetrical.

In chapter 4, we propose to quantify contacts adaptively, by combining the contact graph approach and our newly established upper-bound of D2D goodput approach. Nonetheless, we extended our results to other types of signal propagation and modulation schemes to reveal potential differences the contact characterization.

The fixed contact approach has a volume distribution, which is precisely the same as the contact time distribution, multiplied by the chosen fixed rate. On the other hand, when considering a realistic adaptive contact strategy, the distribution of contact volume is entirely different. More specifically, adaptive-rate contacts have a very high likelihood of having a volume close to 0, which results in a reverse J-shape distribution of contact volume.

As it turns out, the number of small-duration contacts far exceeds the number of long-duration contacts. In terms of actual volume, we found that whether we consider a realistic

modulation scheme (where the throughput goes toward 0) and an unrealistic one (the last modulation remains constant until devices fall out of range), in both cases the majority of the total volume of data exchanged happens using the last modulation.

Because of the behavior of small contacts happening at long distances, considering a fixed-rate will overestimate the volume of contacts. Nonetheless, if one requires a fixed-rate estimate for technical reasons, we propose OpportunistiKapacity, our open-source software for adaptive-rate contact volume calculation, from which one could trivially estimate the best-suited fixed-rate instead of a rough estimate as done in the literature.

Takeaway 3

We shed a new light on short-duration contacts which are, in fact, the backbone of contact networks.

5.2 Perspectives

With this thesis, we opened up several promising leads to gain further insights into D2D and contact graphs. The following includes a list of topics left open for research:

Spatio-temporal variations in geographical centrality estimations. While we showed that we could pin a location (area) to a centrality estimate, the accuracy, and overall reliability may be explored in several different aspects. For instance, the shape of the area was chosen in the form of a grid, but perhaps other solutions such as the Voronoi tessellation could enhance the results significantly. On top of this, we observed the strong correlation between time of day and the success in estimating centralities; studying the geographical-centrality periodicity could help predict upcoming values. Last but not least, with the help of GPS trace over the span of several days, we could implement a database, based on an offline learning of a few days, that the locations are redundant thus definitely solving the need to approximate the centrality of individuals perpetually. Ultimately, this could have a variety of application in mobile offloading through modern routing and dissemination algorithms.

D2D asymmetrical performance-oriented direct graph approach for contact networks. In case of a contact network with heterogeneous devices, which will happen in a real-life scenario, there will be an asymmetrical relationship between the devices. In this thesis, we reveal that this asymmetric feature surprisingly extends to the goodput between two devices; thus, the goodput estimation depends on which way the data is going. First, a subtle characterization of devices D2D performance through their technical specification is required; this includes the hardware, software (OS, D2D API used). Second, by applying the newly defined D2D link performance onto a direct graph, we will be able to quantify contact graphs using a novel D2D performance-based metric. Additionally, we could also model the neighboring interference using the number of links and model the noise through a carefully design a random variable for the noise/shadowing phenomena. Then, using our previously insights gained from the geo-centrality methodology, we could try to see match this newly established metric to a geographical position. Hopefully, this centrality is stable in some way which would then solve the offloading candidate choice in the most data-efficient manner.

References

[1] AlGhamdi, Z., Jamour, F., Skiadopoulos, S., & Kalnis, P. (2017). A Benchmark for Betweenness Centrality Approximation Algorithms on Large Graphs. In *Proceedings of the 29th International Conference on Scientific and Statistical Database Management - SSDBM '17* (pp. 1–12). New York, New York, USA: ACM Press.

[2] Ali, K., Nguyen, H. X., Shah, P., & Vien, Q.-T. (2016). Energy efficient and scalable D2D architecture design for public safety network. In *2016 International Conference on Advanced Communication Systems and Information Security (ACOSIS)* (pp. 1–6).: IEEE.

[3] Ali, K., Nguyen, H. X., Vien, Q. T., Shah, P., & Chu, Z. (2018). Disaster Management Using D2D Communication with Power Transfer and Clustering Techniques. *IEEE Access*, 6, 14643–14654.

[4] Andreev, S., Pyattaev, A., Johnsson, K., Galinina, O., & Koucheryavy, Y. (2014). Cellular traffic offloading onto network-assisted device-to-device connections. *IEEE Communications Magazine*, 52(4), 20–31.

[5] Ansari, R. I., Chrysostomou, C., Hassan, S. A., Guizani, M., Mumtaz, S., Rodriguez, J., & Rodrigues, J. J. P. C. (2018). 5G D2D Networks: Techniques, Challenges, and Future Prospects. *IEEE Systems Journal*, 12(4), 3970–3984.

[6] Awad, A., Frunzke, T., & Dressler, F. (2007). Adaptive distance estimation and localization in WSN using RSSI measures. *Proceedings - 10th Euromicro Conference on Digital System Design Architectures, Methods and Tools, DSD 2007*, (pp. 471–478).

[7] Bader, D. A., Kintali, S., Madduri, K., & Mihail, M. (2007). Approximating Betweenness Centrality. In *Algorithms and Models for the Web-Graph* (pp. 124–137). Berlin, Heidelberg: Springer Berlin Heidelberg.

[8] Bader, D. A., K. S. M. K. M. M. (2007). Approximating betweenness centrality. In *WAW'07 Proceedings of the 5th international conference on Algorithms and models for the web-graph*, volume 4863 LNCS (pp. 124–137).

[9] Baidya, S. & Levorato, M. (2017). Content-based interference management for video transmission in D2D communications underlaying LTE. In *2017 International Conference on Computing, Networking and Communications (ICNC)* (pp. 144–149).: IEEE.

[10] Balasubramanian, A., Levine, B., & Venkataramani, A. (2007). DTN routing as a resource allocation problem. *ACM SIGCOMM Computer Communication Review*, 37(4), 373.

[11] Bao, X., Lin, Y., Lee, U., Rimac, I., & Choudhury, R. R. (2013). DataSpotting: Exploiting naturally clustered mobile devices to offload cellular traffic. In *2013 Proceedings IEEE INFOCOM* (pp. 420–424).: IEEE.

[12] Barthélemy, M. (2011). Spatial networks. *Physics Reports*, 499(1-3), 1–101.

[13] Behrisch, M., Bieker, L., Erdmann, J., & Krajzewicz, D. (2011). SUMO – Simulation of Urban MObility. *Iaria*, (c), 55–60.

[14] Benkic, K., Malajner, M., Planinsic, P., & Cucej, Z. (2008). Using RSSI value for distance estimation in wireless sensor networks based on ZigBee. *Proceedings of the 15th International Conference on Systems, Signals and Image Processing*, (pp. 303–306).

[15] Bentley Systems Incorporated (2018). Legion Software.

[16] Bergamini, E., Meyerhenke, H., & Staudt, C. L. (2014). Approximating Betweenness Centrality in Large Evolving Networks.

[17] Bertenyi, B. (2015). *3GPP system standards heading into the 5G era*. Technical report.

[18] Braha, D. & Bar-Yam, Y. (2006). From centrality to temporary fame: Dynamic centrality in complex networks. *Complexity*, 12(2), 59–63.

[19] BRANDES, U. & PICH, C. (2007). CENTRALITY ESTIMATION IN LARGE NETWORKS. *International Journal of Bifurcation and Chaos*, 17(07), 2303–2318.

[20] Burgess, J., Gallagher, B., Jensen, D., & Levine, B. N. (2006). MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks. In *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*, volume 00 (pp. 1–11).: IEEE.

[21] Camps-Mur, D., Garcia-Saavedra, A., & Serrano, P. (2013). Device-to-device communications with Wi-Fi Direct: overview and experimentation. *IEEE Wireless Communications*, 20(3), 96–104.

[22] Casteigts, A., Flocchini, P., Quattrociocchi, W., & Santoro, N. (2010). Time-Varying Graphs and Dynamic Networks. *International Journal of Parallel, Emergent and Distributed Systems*, 27(5), 387–408.

[23] Chou, C.-M., Li, C.-Y., Chien, W.-M., & Lan, K.-c. (2009). A Feasibility Study on Vehicle-to-Infrastructure Communication: WiFi vs. WiMAX. In *2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware* (pp. 397–398).: IEEE.

[24] Chowdhury, H., Lehtomäki, J., Mäkelä, J.-P., & Kota, S. (2010). Data Downloading on the Sparse Coverage-Based Wireless Networks. *Journal of Electrical and Computer Engineering*, 2010, 1–7.

[25] Cisco (2014). *802.11ac: The Fifth Generation of Wi-Fi*. Technical Report March.

[26] Cisco (2017). *Cisco Visual Networking Index : Global Mobile Data Traffic Forecast Update , 2016 – 2021*. Technical report.

[27] Cisco (2019). *Cisco visual networking index (VNI) global mobile data traffic forecast update, 2017-2022 white paper*. Technical report.

[28] Codeca, L., Frank, R., & Engel, T. (2015). Luxembourg SUMO Traffic (LuST) Scenario: 24 hours of mobility for vehicular networking research. In *2015 IEEE Vehicular Networking Conference (VNC)*, volume 2016-Janua (pp. 1–8). Kyoto, Japan: IEEE.

[29] Cohen, E., Delling, D., Pajor, T., & Werneck, R. F. (2014). Computing classic closeness centrality, at scale. In *Proceedings of the second edition of the ACM conference on Online social networks - COSN '14* (pp. 37–50). New York, New York, USA: ACM Press.

[30] Conti, M. & Giordano, S. (2014). Mobile ad hoc networking: milestones, challenges, and new research directions. *IEEE Communications Magazine*, 52(1), 85–96.

[31] Crucitti, P., Latora, V., & Porta, S. (2006). Centrality measures in spatial networks of urban streets. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 73(3), 1–5.

[32] Daly, E. M. & Haahr, M. (2007). Social network analysis for routing in disconnected delay-tolerant MANETs. In *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing - MobiHoc '07* (pp.32). New York, New York, USA: ACM Press.

[33] Daly, E. M. & Haahr, M. (2009). Social network analysis for information flow in disconnected delay-tolerant MANETs. *IEEE Transactions on Mobile Computing*, 8(5), 606–621.

[34] Deepak G. C., Ladas, A., & Politis, C. (2019). Robust Device-to-Device 5G Cellular Communication in the Post-Disaster Scenario. In *2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)* (pp. 1–6).: IEEE.

[35] Dzurovcak, D. X. & Yang, S. (2017). Performance Analysis of Routing Protocols in Delay Tolerant Networks. In *2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)* (pp. 511–515).: IEEE.

[36] El-Rabbany, A. (2002). *Introduction to GPS: the global positioning system*. Boston, MA: Artech house.

[37] Freeman, L. C. (1977). A Set of Measures of Centrality Based on Betweenness. *Sociometry*, 40(1), 35.

[38] Funai, C., Tapparello, C., & Heinzelman, W. (2017). Enabling multi-hop ad hoc networks through WiFi Direct multi-group networking. In *2017 International Conference on Computing, Networking and Communications (ICNC)* (pp. 491–497).: IEEE.

[39] Furno, A., Faouzi, N. E. E., Sharma, R., & Zimeo, E. (2017). Two-level clustering fast betweenness centrality computation for requirement-driven approximation. In *2017 IEEE International Conference on Big Data (Big Data)*, volume 2018-Janua (pp. 1289–1294).: IEEE.

[40] Goldman, D. (2019). 5G networks are here: Here's everything you need to know.

[41] González, M. C., Hidalgo, C. A., & Barabási, A.-L. (2008). Understanding individual human mobility patterns. *Nature*, 453(7196), 779–782.

[42] Google. ADB.

[43] Google. Nearby - Connections API Overview.

[Google] Google. Nearby Connections - Strategy.

[45] Google. Wi-Fi peer-to-peer overview.

[46] Google. WifiManager.

[47] Grossglauser, M. & Tse, D. (2002). Mobility increases the capacity of ad hoc wireless networks. *IEEE/ACM Transactions on Networking*, 10(4), 477–486.

[48] Guan, L., Zhang, X., Liu, Z., Huang, Y., Lan, R., & Wang, W. (2013). Spatial modeling and analysis of traffic distribution based on real data from current mobile cellular networks. *Proc. of ICCP'13 - 2013 IEEE International Conference on Computational Problem-Solving*, (pp. 135–138).

[49] Guimera, R., Mossa, S., Turtschi, A., & Amaral, L. A. N. (2005). The worldwide air transportation network: Anomalous centrality, community structure, and cities' global roles. *Proceedings of the National Academy of Sciences*, 102(22), 7794–7799.

[50] Helgason, O., Kouyoumdjieva, S. T., & Karlsson, G. (2014). Opportunistic Communication and Human Mobility. *IEEE Transactions on Mobile Computing*, 13(7), 1597–1610.

[51] Hom, J., Good, L., & Shuhui Yang (2017). A survey of social-based routing protocols in Delay Tolerant Networks. In *2017 International Conference on Computing, Networking and Communications (ICNC)* (pp. 788–792).: IEEE.

[52] Hui, P., Crowcroft, J., & Yoneki, E. (2011). BUBBLE Rap: Social-Based Forwarding in Delay-Tolerant Networks. *IEEE Transactions on Mobile Computing*, 10(11), 1576–1589.

[53] Igarashi, Y. & Miyazaki, T. (2018). A DTN routing algorithm adopting the "Community" and "Centrality" parameters used in social networks. In *2018 International Conference on Information Networking (ICOIN)*, volume 2018-Janua (pp. 211–216).: IEEE.

[54] Jameel, F., Faisal, Haider, M. A. A., & Butt, A. A. (2017). Robust localization in wireless sensor networks using RSSI. In *13th International Conference on Emerging Technologies (ICET)*, volume 2018-Janua (pp. 1–6).: IEEE.

[55] Jameel, F., Hamid, Z., Jabeen, F., Zeadally, S., & Javed, M. A. (2018). A Survey of Device-to-Device Communications: Research Issues and Challenges. *IEEE Communications Surveys & Tutorials*, 20(3), 2133–2168.

[56] Kang, U., Papadimitriou, S., Sun, J., & Tong, H. (2011). Centralities in Large Networks: Algorithms and Observations. In *Proceedings of the 2011 SIAM International Conference on Data Mining* (pp. 119–130). Philadelphia, PA: Society for Industrial and Applied Mathematics.

[57] Keller, L., Le, A., Cici, B., Seferoglu, H., Fragouli, C., & Markopoulou, A. (2012). MicroCast: Cooperative Video Streaming on Smartphones. In *Proceedings of the 10th international conference on Mobile systems, applications, and services - MobiSys '12*, volume 68 (pp.57). New York, New York, USA: ACM Press.

[58] Keränen, A. & Ott, J. (2007). Increasing reality for DTN protocol simulations. *Helsinki University of Technology, Tech. Rep*.

[59] Kim, C.-M., Han, Y.-H., Youn, J.-S., & Jeong, Y.-S. (2014). A Socially Aware Routing Based on Local Contact Information in Delay-Tolerant Networks. *The Scientific World Journal*, 2014, 1–7.

[60] Kim, H., Tang, J., Anderson, R., & Mascolo, C. (2012). Centrality prediction in dynamic human contact networks. *Computer Networks*, 56(3), 983–996.

[61] Koetsier, J. (2019). Hong Kong Protestors Using Mesh Messaging App China Can't Block: Usage Up 3685%. *Forbes*.

[62] Kuruvatti, N. P., Klein, A., Ji, L., Zhou, C., Bulakci, O., Eichinger, J., Sattiraju, R., & Schotten, H. D. (2015). Robustness of Location Based D2D Resource Allocation against Positioning Errors. In *IEEE 81st Vehicular Technology Conference (VTC Spring)*, volume 2015 (pp. 1–6).: IEEE.

[63] Lee, K., Rhee, I., Lee, J., Chong, S., & Yi, Y. (2010). Mobile data offloading. In *Proceedings of the 6th International COnference on - Co-NEXT '10*, volume 21 (pp.1). New York, New York, USA: ACM Press.

[64] Lee, M.-J., Lee, J., Park, J. Y., Choi, R. H., & Chung, C.-W. (2012). QUBE: a quick algorithm for updating betweenness centrality. In *Proceedings of the 21st international conference on World Wide Web - WWW '12* (pp. 351). New York, New York, USA: ACM Press.

[65] Li, M., Liao, W., Chen, X., Sun, J., Huang, X., & Li, P. (2018). Economic-Robust Transmission Opportunity Auction for D2D Communications in Cognitive Mesh Assisted Cellular Networks. *IEEE Transactions on Mobile Computing*, 17(8), 1806–1819.

[66] Li, N. & Gillet, D. (2013). Identifying influential scholars in academic social media platforms. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining - ASONAM '13* (pp. 608–614). New York, New York, USA: ACM Press.

[67] Lima, A. & Musolesi, M. (2012). Spatial dissemination metrics for location-based social networks. *Proceedings of the 2012 ACM Conference on Ubiquitous Computing - UbiComp '12*, (pp. 972).

[68] Lindgren, A., Doria, A., & Schelén, O. (2003). Probabilistic routing in intermittently connected networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 7(3), 19.

[69] Lui, G., Gallagher, T., Li, B., Dempster, A. G., & Rizos, C. (2011). Differences in RSSI readings made by different Wi-Fi chipsets: A limitation of WLAN localization. *2011 International Conference on Localization and GNSS, ICL-GNSS 2011*, (pp. 53–57).

[70] Maccari, L., Ghiro, L., Guerrieri, A., Montresor, A., & Cigno, R. L. (2018). On the Distributed Computation of Load Centrality and its Application to DV Routing. In *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, volume 2018-April (pp. 2582–2590). Honolulu, HI: IEEE.

[71] Magaia, N., Francisco, A. P., Pereira, P., & Correia, M. (2015). Betweenness centrality in Delay Tolerant Networks: A survey. *Ad Hoc Networks*, 33, 284–305.

[72] Maglaras, L. A. & Katsaros, D. (2012). New measures for characterizing the significance of nodes in wireless ad hoc networks via localized path-based neighborhood analysis. *Social Network Analysis and Mining*, 2(2), 97–106.

[73] Marsden, P. V. (2002). Egocentric and sociocentric measures of network centrality. *Social Networks*, 24(4), 407–422.

[74] Martín-Campillo, A., Crowcroft, J., Yoneki, E., & Martí, R. (2013). Evaluating opportunistic networks in disaster scenarios. *Journal of Network and Computer Applications*, 36(2), 870–880.

[75] Moghaddam, J. Z., Usman, M., & Granelli, F. (2018). A Device-to-Device Communication-Based Disaster Response Network. *IEEE Transactions on Cognitive Communications and Networking*, 4(2), 288–298.

[76] Neto, J. B. P., Mota, E., Almeida, P., & Rojas, R. (2012). A model for contact volume prediction in DTNs. In *2012 IEEE Symposium on Computers and Communications (ISCC)* (pp. 000199–000202).: IEEE.

[77] Newville, M. & Stensitzki, T. (2017). Non-Linear Least-Squares Minimization and Curve-Fitting for Python Matthew Newville, Till Stensitzki, and others.

[78] Nunes, I. O., Celes, C., Nunes, I., Vaz de Melo, P. O. S., & Loureiro, A. A. F. (2018). Combining Spatial and Social Awareness in D2D Opportunistic Routing. *IEEE Communications Magazine*, 56(1), 128–135.

[79] Okamoto, K., Chen, W., & Li, X.-Y. (2008). Ranking of Closeness Centrality for Large-Scale Social Networks. In *Frontiers in Algorithmics* (pp. 186–195). Berlin, Heidelberg: Springer Berlin Heidelberg.

[80] Passarella, A. & Conti, M. (2013). Analysis of Individual Pair and Aggregate Intercontact Times in Heterogeneous Opportunistic Networks. *IEEE Transactions on Mobile Computing*, 12(12), 2483–2495.

[81] Pedapolu, P. K., Kumar, P., Harish, V., Venturi, S., Bharti, S. K., Kumar, V., & Kumar, S. (2017). Mobile Phone User's Speed Estimation using WiFi Signal-to-Noise Ratio. In *Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing - Mobihoc '17*, number 1 (pp. 1–2). New York, New York, USA: ACM Press.

[82] Pu, L., Chen, X., Xu, J., & Fu, X. (2016). D2D Fogging: An Energy-Efficient and Incentive-Aware Task Offloading Framework via Network-assisted D2D Collaboration. *IEEE Journal on Selected Areas in Communications*, 34(12), 3887–3901.

[83] Pyattaev, A., Johnsson, K., Andreev, S., & Koucheryavy, Y. (2015). Communication challenges in high-density deployments of wearable wireless devices. *IEEE Wireless Communications*, 22(1), 12–18.

[84] Qayyum, S., Shahriar, M., Kumar, M., & Das, S. K. (2013). PCV: Predicting contact volume for reliable and efficient data transfers in opportunistic networks. *Proceedings - Conference on Local Computer Networks, LCN*, (pp. 801–809).

[85] Raffelsberger, C. & Hellwagner, H. (2012). Evaluation of MANET Routing Protocols in a Realistic Emergency Response Scenario. In *Proceedings of the 10th International Workshop on Intelligent Solutions in Embedded Systems* (pp. 88–92).

[86] Rappaport, T. (2001). Wireless Communications: Principles and Practice.

[87] Rebecchi, F., Dias de Amorim, M., Conan, V., Passarella, A., Bruno, R., & Conti, M. (2015). Data Offloading Techniques in Cellular Networks: A Survey. *IEEE Communications Surveys & Tutorials*, 17(2), 580–603.

[88] Reina, D. G., Askalani, M., Toral, S. L., Barrero, F., Asimakopoulou, E., & Bessis, N. (2015). A Survey on Multihop Ad Hoc Networks for Disaster Response Scenarios. *International Journal of Distributed Sensor Networks*, 2015, 1–16.

[89] Riondato, M. & Kornaropoulos, E. M. (2016). Fast approximation of betweenness centrality through sampling. *Data Mining and Knowledge Discovery*, 30(2), 438–475.

[90] Ryoo, J., Kim, H., & Das, S. R. (2012). Geo-fencing: Geographical-fencing based energy-aware proactive framework for mobile devices. In *2012 IEEE 20th International Workshop on Quality of Service* (pp. 1–9).

[91] Saha, S. K., Deshpande, P., Inamdar, P. P., Sheshadri, R. K., & Koutsonikolas, D. (2015). Power-throughput tradeoffs of 802.11n/ac in smartphones. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, volume 26 (pp. 100–108).: IEEE.

[92] Sarafijanovic-Djukic, N., Pidrkowski, M., & Grossglauser, M. (2006). Island Hopping: Efficient Mobility-Assisted Forwarding in Partitioned Networks. In *2006 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, volume 1 (pp. 226–235).: IEEE.

[93] Schurgot, M. R., Comaniciu, C., & Jaffres-Runser, K. (2012). Beyond traditional DTN routing: social networks for opportunistic communication. *IEEE Communications Magazine*, 50(7), 155–162.

[94] Shah, S. T., Hasan, S. F., Seet, B.-C., Chong, P. H. J., & Chung, M. Y. (2018). Device-to-Device Communications: A Contemporary Survey. *Wireless Personal Communications*, 98(1), 1247–1284.

[95] Shevade, U., Han Hee Song, Lili Qiu, & Zhang, Y. (2008). Incentive-aware routing in DTNs. In *2008 IEEE International Conference on Network Protocols* (pp. 238–247).: IEEE.

[96] Singh, B. K. & Gupte, N. (2005). Congestion and decongestion in a communication network. *Physical Review E*, 71(5), 055103.

[97] Sommer, C., Joerer, S., & Dressler, F. (2012). On the applicability of Two-Ray path loss models for vehicular network simulation. In *2012 IEEE Vehicular Networking Conference (VNC)* (pp. 64–69).: IEEE.

[98] Soorki, M. N., Saad, W., Manshaei, M. H., & Saidi, H. (2019). Social Community-Aware Content Placement in Wireless Device-to-Device Communication Networks. *IEEE Transactions on Mobile Computing*, 18(8), 1938–1950.

[99] Spyropoulos, T., Psounis, K., & Raghavendra, C. S. (2005). Spray and wait. In *Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking - WDTN '05* (pp. 252–259). New York, New York, USA: ACM Press.

[100] Strano, E., Cardillo, A., Iacoviello, V., Latora, V., Messora, R., Porta, S., & Scellato, S. (2007). Street centrality vs. commerce and service locations in cities: a Kernel Density Correlation case study in Bologna, Italy. *arXiv preprint physics/0701111*, (pp. 1–14).

[101] Thilakarathna, K., Viana, A. C., Seneviratne, A., & Petander, H. (2017). Design and Analysis of an Efficient Friend-To-Friend Content Dissemination System. *IEEE Transactions on Mobile Computing*, 16(3), 702–715.

[102] Trifunovic, S., Kouyoumdjieva, S. T., Distl, B., Pajevic, L., Karlsson, G., & Plattner, B. (2017). A Decade of Research in Opportunistic Networks: Challenges, Relevance, and Future Directions. *IEEE Communications Magazine*, 55(1), 168–173.

[103] Uppoor, S., Trullols-Cruces, O., Fiore, M., & Barcelo-Ordinas, J. M. (2014). Generation and Analysis of a Large-Scale Urban Vehicular Mobility Dataset. *IEEE Transactions on Mobile Computing*, 13(5), 1061–1075.

[104] Vo, N. S., Duong, T. Q., Tuan, H. D., & Kortun, A. (2018). Optimal Video Streaming in Dense 5G Networks With D2D Communications. *IEEE Access*, 6, 209–223.

[105] von Ferber, C., Holovatch, T., Holovatch, Y., & Palchykov, V. (2009). Public transport networks: empirical analysis and modeling. *The European Physical Journal B*, 68(2), 261–275.

[106] Wang, J., Jiang, C., Bie, Z., Quek, T. Q., & Ren, Y. (2016a). Mobile Data Transactions in Device-To-Device Communication Networks: Pricing and Auction. *IEEE Wireless Communications Letters*, 5(3), 300–303.

[107] Wang, Y., Li, Y., Wang, W., & Song, M. (2016b). A Locality-Based Mobile Caching Policy for D2D-Based Content Sharing Network. In *2016 IEEE Global Communications Conference (GLOBECOM)* (pp. 1–6).: IEEE.

[108] Wenjing Wang, Fei Xie, & Chatterjee, M. (2009). Small-Scale and Large-Scale Routing in Vehicular Ad Hoc Networks. *IEEE Transactions on Vehicular Technology*, 58(9), 5200–5213.

[109] Whitbeck, J., Amorim, M., Lopez, Y., Leguay, J., & Conan, V. (2011). Relieving the wireless infrastructure: When opportunistic networks meet guaranteed delays. *2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM 2011 - Digital Proceedings*.

[110] Wu, H., Fujimoto, R., Guensler, R., & Hunter, M. (2004). MDDV. In *Proceedings of the first ACM workshop on Vehicular ad hoc networks - VANET '04* (pp.47). New York, New York, USA: ACM Press.

[111] Xu, J., Liu, W., Lang, F., Zhang, Y., & Wang, C. (2010). Distance Measurement Model Based on RSSI in WSN. *Wireless Sensor Network*, 02(08), 606–611.

[112] Yong Ding & Li Xiao (2010). SADV: Static-Node-Assisted Adaptive Data Dissemination in Vehicular Networks. *IEEE Transactions on Vehicular Technology*, 59(5), 2445–2455.

[113] Zeng, Y., Pathak, P. H., & Mohapatra, P. (2014). A first look at 802.11ac in action: Energy efficiency and interference characterization. In *2014 IFIP Networking Conference*, volume 20 (pp. 1–9).: IEEE.

[114] Zhang, D. & Sterbenz, J. P. (2015). Robustness analysis of mobile ad hoc networks using human mobility traces. In *2015 11th International Conference on the Design of Reliable Communication Networks (DRCN)* (pp. 125–132).: IEEE.

[115] Zhao, G., Chen, S., Qi, L., Zhao, L., & Hanzo, L. (2019). Mobile-Traffic-Aware Offloading for Energy- and Spectral-Efficient Large-Scale D2D-Enabled Cellular Networks. *IEEE Transactions on Wireless Communications*, 18(6), 3251–3264.

[116] Zhao, J., Zhang, Y., & Cao, G. (2007). Data pouring and buffering on the road: A new data dissemination paradigm for vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology*, 56(6 I), 3266–3277.

[117] Zhao, W., Chen, Y., Ammar, M., Corner, M., Levine, B., & Zegura, E. (2006). Capacity Enhancement using Throwboxes in DTNs. In *2006 IEEE International Conference on Mobile Ad Hoc and Sensor Systems* (pp. 31–40). Florence, Italy: IEEE.

[118] Zhu, X., Li, Y., Jin, D., & Lu, J. (2017). Contact-Aware Optimal Resource Allocation for Mobile Data Offloading in Opportunistic Vehicular Networks. *IEEE Transactions on Vehicular Technology*, 66(8), 7384–7399.

[119] Zhu, Y., Xu, B., Shi, X., & Wang, Y. (2013). A survey of social-based routing in delay tolerant networks: Positive and negative social effects. *IEEE Communications Surveys and Tutorials*, 15(1), 387–401.

A

Realistic contact computation library: OpportunistiKapacity

We provide a python library to turn mobility traces into opportunistic contact capacity. This library aims to simplify the calculation of total network data capacity, more precisely in the context of contact networks.

A.1 Basic run using wrapper

You must provide at least 4 parameters for the software to run:

1. The trace kind
2. The trace path
3. A propagation model
4. A modulation scheme

If these arguments are known, you can run the script:

```
./OKwrapper.py trace_kind dataset propagation_model modulation_scheme
```

A.2 Supported traces

This library only considers two types of datasets. Both are text-based.

A.3 Mobility trace

The *mobility* trace kind is a format where all existing nodes have their geographical coordinates updated at each snapshot. For example:

time	node	x	y
0.0	a	50479.67	54248.598
0.0	b	47721.375	54930.71
0.0	c	48519.473	55673.586
0.0	d	48798.016	54503.477
0.0	e	50233.27	55252.617
1.0	a	50479.684	54248.62
1.0	b	47722.023	54930.7
1.0	c	48518.676	55675.266
1.0	d	48798.703	54505.332
1.0	e	50235.68	55251.816

Table A.1: Snippet from a mobility trace.

Publishing traces in such a format is a common practice within mobility simulation studies (e.g., [TAPASCologne](http://kolntrace.project.citi-lab.fr/)).

A.3.1 Contact trace

The *contact* trace kind is a format describing a contact duration between two nodes, regardless of when the contact actually happened.

The above snippet has four columns, from left to right: id node 1, id node 2, start contact, end contact.

node 1	node 2	beginning contact	end contact
1	2	1156084891	1156084895
2	3	1156085092	1156085159
4	2	1156085110	1156085118
5	8	1156085190	1156085191
4	2	1156085219	1156085302

Table A.2: Snippet from a contact trace.

A.3.2 Propagation models

The propagation models, found in the 'communications' module, calculate the signal loss according to the distance between the receiver and transmitter. So far we include the following path loss models:

- Freespace (`freespace_loss`)
- Log-distance (`logDistance_loss`)
- Two-Ray ground reflection (`twoRay_loss`)

A.3.3 Modulation schemes

The modulation schemes, found in the 'communications' module, set a data rate for a given RSSI between two nodes. So far, we propose the following path loss models (explained thoroughly in the fourth chapter):

- `Wifi5_empirical_goodput`
- `Wifi5_stepwise_max`
- `Wifi5_stepwise_linear_adjusted`
- `Wifi5_stepwise_fit`

A.4 Module and classes

The module is separated in 3 files:

- `communications`: every function to do with the translation of distance into a throughput estimation.
- `datasetparser`: provides two classes, (*ContactParser*, *MobilityParser*) to iterate over a dataset file.
- `contactcalculator`: provides the two classes (*GeographicTrace*, *ContactTrace*) to calculate the contact capacity.