UNIVERSIDAD CARLOS III DE MADRID

TESIS DOCTORAL

Performance Evaluation of Floating Content for Context-aware Applications

Autor:
Shahzad Ali

Directores:
Marco Ajmone Marsan
Gianluca Rizzo
Vincenzo Mancuso

DEPARTAMENTO DE INGENIERÍA TELEMÁTICA

Leganés, Septiembre de 2014

UNIVERSIDAD CARLOS III DE MADRID

Ph.D. THESIS

PERFORMANCE EVALUATION OF FLOATING CONTENT FOR CONTEXT-AWARE APPLICATIONS

Author:
Shahzad Ali

Directors:
Marco Ajmone Marsan
Gianluca Rizzo
Vincenzo Mancuso

DEPARTMENT OF TELEMATIC ENGINEERING

Leganés, September 2014

Dedication

To my parents and loving wife

Acknowledgments

First and foremost I wish to express my sincere thanks to the directors of my thesis. I thank Marco Ajmone Marsan for his invaluable guidance and insightful suggestions, which contributed to the success of my work. I am extremely grateful to Gianluca Rizzo for his continuous support of my Ph.D. study, tremendous source of guidance and for his motivation, invaluable feedback and patience to me.

I am also extremely grateful to Vincenzo Mancuso and Balaji Rengarajan for teaching me how to tackle the most difficult problems, develop self-confidence and always be determined to overcome the obstacles in my research.

I also want to thank collectively all the friends and colleagues from IMDEA Networks for making my stay in Madrid an unforgettable experience.

Abstract

Context-awareness is a peculiar characteristic of an expanding set of applications that make use of a combination of restricted spatio-temporal locality and mobile communications, to deliver a variety of services. Opportunistic communications satisfy well the communication requirements of these applications, because they naturally incorporate context. Recently, an opportunistic communication paradigm called "Floating Content" was proposed, to support infrastructure-less, distributed content sharing. It aims at ensuring the availability of data within a certain geographic area called "anchor zone". In literature, the focus was on understanding the asymptotic properties of the floating lifetime, i.e., the duration of time for which content floats in the anchor zone. Instead, our objective is to characterize the performance of context-aware applications using floating content as a communication service.

First, we present a simple approximate analytical model for accessing the viability of floating content to act as a communication service for context-aware applications. We focus on the "success probability", which captures the likelihood for a user to receive the content when traversing the anchor zone and apply our analysis to estimate the success probability for three representative categories of context-aware applications, and show how the system can be configured to achieve the application's target.

Second, we investigate the impact of different mobility models on the performance of context-aware applications that use floating content. In particular, we consider four different mobility models, and, by using extensive simulation experiments, we investigate the performance of three different categories of context-aware applications. By comparing the simulation results to the performance predictions of our previously proposed simple analytical model, we show that our model can provide useful performance predictions even for complex and realistic mobility models. Simulation results under different mobility models also confirm the viability of floating content to act as a communication service for a variety of context-aware applications.

Finally, we investigate the performance of floating content in a real world setting by developing and deploying an Android mobile application based on floating content in

an office and a university campus environment. To the best of our knowledge, this is the first ever experimental evaluation of floating content service in a real setting. Our results provide quite interesting indications for the viability and the implementation of applications using floating content in the considered environments. We also propose a novel simple analytical model that accounts for the peculiarities of the mobility patterns in such a real world setting, and that can accurately predict the effectiveness of floating content for the implementation of context-aware applications in an office and a campus setting.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The growth of mobile computing, and the pervasiveness of smart user devices is progressively driving applications towards context-awareness, i.e., towards applications and services that allow users to exploit "any information that can be used to characterize the situation of an entity" [1]. This enables applications to improve their efficiency and utility, and to offer services that better suit the needs of users in a given situation [2].

One of the best examples of context, and one that is widely used by context-aware applications is spatial and temporal locality [3–6]. As an example, consider a context-aware parking finding application [7]. Information about a vacant parking spot may be of interest for a limited time (until the space is filled), and only to users who are in fairly close proximity. Similarly, a shop might want to spread advertisements about a sale only during the sale period and to customers in the neighborhood. Clearly, the characteristics of such types of information are radically different from those of, e.g., generic data files, which corresponds to the majority of traffic carried by the Internet today. Many more examples of context-aware applications are emerging, that make use of spatio-temporal locality and wireless communications to deliver a variety of services [8–16]. By the end of year 2013, there were more than 1 million applications developed for the Android based smartphones alone, and about 10% of them employed localization technology [17]. For instance, it is expected that by the end of 2014 more than 1.5 billion people would be using applications based on local search (search restricted on the basis of spatio-temporal locality), and that mobile location based services will drive revenues of more than $\$15$ billion worldwide [7, 18].

A common feature of context-aware applications is that they have communication requirements that significantly differ from ordinary applications. For most location-based, context-aware applications, the scope of generated content itself is local. This locally relevant content may be of little concern to the rest of the world, therefore moving this content from the user device to store it in a well-accessible centralized location and/or making this information available beyond its scope represents a clear waste of resources (connectivity, storage) [19]. Due to these specific requirements, opportunistic communication can play a special role when coupled with context-awareness.

The benefit of opportunistic communications is that it naturally incorporates context as spatial proximity is not only associated to connectivity, but also, at the application layer, to correlation at several levels between communicating peers, between their needs, interests, etc. (the fact that they are in proximity of each other might be because they share interests and views: a same restaurant might mean same tastes for food, etc.). Indeed, connectivity to the infrastructure as a prerequisite is often limiting due to cost and capacity concerns, especially for mobile users for whom using such applications may be problematic due to high roaming charges, unavailability of data services, or simply no network coverage. Moreover devices like smart-phones and PDAs have enabled users to dynamically produce content like pictures and videos resulting in generation of unprecedented amounts of data. Therefore a side effect of it is that cellular networks are getting overloaded leading to deteriorating 3G quality. As a result, cellular network operators like AT&T, T-Mobile, Vodafone, and Orange are already using several data offloading solutions to Wi-Fi networks. Most of the wireless providers using different methods like imposing a data limit per month [20] or offloading data traffic to Wi-Fi networks is a clear indication that existing network operators cannot cope with high demand of the user therefore there is a need to explore other modes of communication.

Several of the recent trends in networking also seem to be pushing the implementation of ICT services closer to the end user. On the one hand, Software Defined Networking (SDN) [5] and the new paradigm of Fog Computing [4] can bring personalized services to end users, accounting for the context in which the user is immersed at any moment, expanding the well-known concept of Location-Based Services [3]. On the other hand, the emergence of the Internet of Things (IoT) and even of the Internet of Everything (IoE) concepts increases connectivity to levels at which the centralization of services, processing and storage will become impossible [6]. This drive towards locality calls for the exploitation of data which are sensed locally, and which are relevant for the user profile, location and context; in addition, it also calls for the exploitation of localized opportunistic terminal-to-terminal infrastructureless communications, to disseminate locally those data which can become irrelevant, if brought to the network core or to other periphery areas, and to offload information transfers with local relevance from the bandwidth-hungry infrastructured wireless access and network core transit. This justifies the growing interest in terminal-to-terminal connectivity and in opportunistic communications. Opportunistic communications allow the users to exchange contents to everyone in their proximity without any limitation as an Internet connection or an account connected to a social network. It can be considered the easiest and fastest way to share information on the fly and probably it is the best way to deal with geographically limited information that have no reason to be advertised on the Web (like a market broadcasting offers and discounts to users passing by).

In our work we consider a specific opportunistic communication paradigm, known as *floating content* (FC) [21], conceived to support server-less distributed content sharing. It aims at ensuring the availability of data within a certain geographic area called *anchor zone* (AZ), and for a given duration in time. Within the AZ, any time a user who is unaware of the content enters the transmission range of another user possessing it, the content is shared. Whenever a node having content exits the AZ, it deletes that content. Thus, the content can be replicated on a set of nodes within the AZ, who in turn will pass it on to other nodes that come into their range before leaving the AZ. As a result, information might be stored on some nodes within the AZ even after the original node which generated it has left, i.e., content 'floats' within the AZ. Users who traverse the AZ while content is floating have an opportunity to learn the content, provided they meet a node with content prior to leaving the AZ.

In the literature [21–23], the focus is on evaluating the general feasibility of FC and understanding the asymptotic properties of the floating lifetime. In contrast, our objective is to characterize the performance of context-aware applications that use FC as a communication service. Indeed, from an application perspective, it is not sufficient that the content asymptotically floats: what matters is the probability for a node traversing the AZ to get the content which is floating. In our work, we study the *success probability*, a key performance indicator that captures the likelihood of intended users to receive the relevant information. For understanding the impact of system design parameters of FC like AZ radius, node transmission range, and node density on success probability, we develop a simple approximate analytical model and demonstrate how the model predictions can be used to tune key system parameters of FC to achieve the desired application performance. Then we validate our analysis with extensive simulations under quite diverse mobility model showing that FC is capable of providing quite reasonable performance for context-aware applications in terms of success probability. As a final step for performance evaluation, we conducted real world experiments by developing and deploying an Android mobile application based on FC in an office and a university campus environment. We also propose a new analytical model which accurately captures the key behaviors that emerged from the experiments.

1.1 Summary of Thesis Contributions

The contributions of this thesis are summarized as follows. First, we develop a simple (in that it uses few primitive system parameters), approximate analytical model for the performance analysis of context-aware applications that use FC as the communication service. In contrast to the previous modeling studies of FC [21, 22], our objective is to characterize the performance of context-aware applications relying on the FC service. We focus on the success probability, which captures the likelihood for a user to receive the content when traversing the AZ. From a system design perspective, our analysis can be used to tune key system parameters so as to achieve the desired application performance. A key characteristic of our modeling approach is that success probability is computed from few primitive system parameters allowing our analysis to be generalized to various settings, including different AZ shapes, different user mobility patterns, different user speed distributions, different service and application models. We apply our analysis to estimate the success probability for three representative categories of context-aware applications, and show how the system can be configured to achieve the application's target. In order to complement our analytical study, we validate our model using extensive simulations under different settings. Our simulation results show that our model-based predictions are indeed highly accurate under a wide range of conditions.

Second, we investigate the impact of different mobility models on the performance of FC. In particular, we consider four different mobility models, and, by using extensive simulation experiments, we investigate the performance of three different categories of context-aware applications that use FC. We also compare the simulation results to the performance predictions of our previously proposed simple analytical model showing that our model can provide useful performance predictions even for complex and realistic mobility models. Simulation results under different mobility models also confirm the viability of FC to act as a communication service for a variety of context-aware applications.

Our studies of the performance of FC and of the effect that system parameters, such as mobility and node density, have on it, have been based on analytical and/or simulation models. All these pieces of work are based on simplifying assumptions for user mobility, and on a simplified model for data exchange. Therefore, they still leave open the issue of determining in realistic settings what are the range of operating conditions (which parts of a city or of a building, what kind of mobility pattern, etc.) in which FC is able to support context-aware applications with a given performance target. Therefore as the third contribution, we investigate the performance of FC in a real world setting by developing and deploying an Android mobile application based on FC in an office environment. To the best of our knowledge, this is the first ever experimental evaluation of FC service in an office setting. Our results confirm the suitability of FC as a communication service for context-aware applications in an office setting.

Finally, we do the first ever experimental evaluation of the FC communication paradigm in a university campus setting. From the experiment, we characterize the mobility patterns, and we assess the performance of the applications implemented using the FC paradigm. Our results unveil the key relevance of group dynamics in user movements for the FC performance. Surprisingly, in such an environment, our results show that a relatively low user density is enough to guarantee content persistence over time, contrarily to predictions from available models. We also investigate the impact of multiple seeders on the performance and we show that by increasing the number of seeders in the network, it is easy to achieve huge performance improvement. Based on these experimental findings, we develop a novel simple analytical model that accounts for the peculiarities of the mobility patterns in such a setting, and that can accurately predict the effectiveness of FC for the implementation of services in a campus/large office setting.

1.2 Thesis Overview

The thesis is structured as follows. In Chapter 2, we explain the basic operation of FC service and discuss the most relevant related works emphasizing on the differences with respect to our work. In Chapter 3, we present a simple approximate analytical model for the performance analysis of context-aware applications that use FC. In Chapter 4, we investigate the impact of different mobility models on the performance of context-aware applications using FC and show how well FC behaves in realistic mobility settings, and how closely the values of success probability predicted by our simple analytical model match those obtained with complex mobility models. In Chapter 5, we analyze the performance of FC service in a real world environment by developing and deploying an Android mobile application in an office environment. In Chapter 6, we present an experimental study of the operation of FC in a university campus context by characterizing some critical performance aspects which are relevant for applications based on FC. We also propose a new analytical model which accurately captures the key behaviors that emerged from the experiment. Finally in Chapter 7, we conclude by discussing the implications that result from this thesis and future research directions.

1.3 Research Publications from Thesis

The following research papers were published during the course of Ph.D:

- S. Ali, G. Rizzo, B. Rengarajan, and M. A. Marsan, "A simple approximate analysis of floating content for context-aware applications," in *Proceedings of the fourteenth ACM international symposium on Mobile ad hoc networking and computing, ser.* MobiHoc 13. New York, NY, USA: ACM, 2013, pp. 271-276.
- S. Ali, G. Rizzo, M. Ajmone Marsan, and V. Mancuso, "Impact of mobility on the performance of context-aware applications using floating content," in *Context-Aware Systems and Applications*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, P. C. Vinh, V. Alagar, E. Vassev, and A. Khare, Eds. Springer International Publishing, 2014, vol. 128, pp. 198-208.
- S. Ali, G. Rizzo, V. Mancuso, V. Cozzolino, and M. Ajmone Marsan, "Experimenting with floating content in an office setting," *Communications Magazine, IEEE*, vol. 52, no. 6, pp. 49-54, June 2014.

The following research paper has been submitted and is going through review process:

- S. Ali, G. Rizzo, M. Ajmone Marsan, and V. Mancuso, "Persistence and availability of floating content in a campus environment," submitted to *INFOCOM 2015*.

Chapter 2

Background

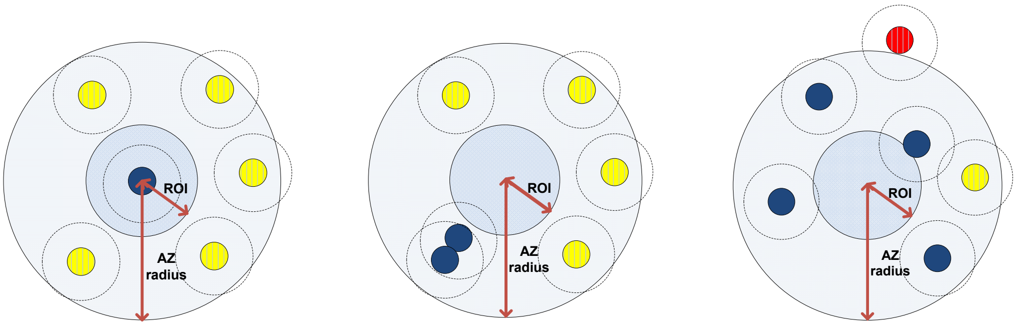
In this chapter we first summarize the basic operation of floating content service and then discuss the most relevant related works emphasizing differences with respect to our work.

2.1 Floating Content Service

In this section we describe the basics of the operation of the Floating Content communication service, which we refer to as FC. FC is an information sharing paradigm that allows the implementation of a class of infrastructure-less services based on opportunistic wireless terminal-to-terminal communications.

Fig. 2.1 summarizes FC operation. Each wireless node participating in a given FC service produces content which is of interest for users within a limited geographical area called *anchor zone* (AZ), for a given interval of time. Therefore, content is generated inside the AZ and is destroyed when a node exits the AZ [21]. Whenever a node having a content in the AZ comes within the transmission range of some other node which does not have such content, the content is replicated through opportunistic message exchanges. In this way, contents "float" within the AZ, i.e., they are available on a set of nodes which move within the AZ. The set of nodes carrying each content varies over time, even after the node which generated the content has left the AZ.

Through this geographically constrained opportunistic replication mechanism, a given content is stored probabilistically in a spatial region without the support of fixed infrastructure, and it is made available to users traversing the AZ through opportunistic exchanges with nodes having content in the AZ. Specifically, the theory shows that, within the AZ, a content floats over time with high probability, based on a *criticality* condition, which accounts for the average number of nodes in the AZ, the average node contact



(a) A node generates a content.

(b) The content is replicated opportunistically within the AZ.

(c) Through replication, content is made available in the AZ. Nodes exiting the AZ discard the content.

Figure 2.1: Operation of Floating Content service.

rate, and the average node sojourn time in the AZ [21]. When the criticality condition is satisfied, the expected lifetime of a content is infinite under the fluid limit approximation of [21]. In practice, this implies large average content lifetimes.

However, the sole condition of floating does not provide service guarantees. In fact, a content needs to be *available* for new nodes to obtain it when they enter the AZ, and it needs to reach the nodes transiting through the AZ with suitably high probability [24].

For this reason the performance metric we consider for the FC service is the probability that a user entering the AZ receives the floating content *timely*. We call this parameter the *success probability* of the FC service. The exact definition of this parameter depends on the way we define the time by which a new user who enters the AZ gets the content. The determination of this time is application specific, and is made with reference to a subregion of the AZ called the *Range Of Interest* (ROI), as shown in Fig. 2.1. In particular, a content can be floated into a large area (AZ), but, from an application perspective, content might be needed only within a subset of that large area (ROI). Therefore, the AZ acts as a replication range in which content is replicated using FC, while the ROI is application-specific, and depends on the particular needs of an application. We consider three cases, corresponding to three different categories of context-aware applications, and to three different definitions of success probability:

Baseline application: In this case, ROI and AZ are coincident, and the success probability is the probability that a new user entering in the AZ gets the content before leaving the AZ.

Application category 1: For this category, the message must be delivered to the new user by the time it exits the ROI. Typically, in these applications the message is expected to trigger some specific actions once the user is within the ROI. One example of such application can be advertising, when the fact of traversing a given area makes a user very likely to be interested in a specific offer/discount. For such applications, success probability is defined as the percentage of times a node gets content before exiting the ROI.

Application category 2: For this category of applications, the content must be delivered to users before they enter the ROI. Examples of such applications can be accident or traffic jam warnings, when a user should be notified in time to take informed decisions about alternative paths. Here success probability is the probability of getting the content before entering the ROI.

2.2 Related Work

The concept of Floating Content is not new, and in recent years concepts similar to it have appeared in the literature with different names [25–32].

In [33], a concept very much similar to FC called *hovering information* is presented. The focus of hovering information is sharing content with spatio-temporal locality constraints, using pure ad hoc communications, like FC. The authors of [33] presented two algorithms aiming to improve the availability of a piece of hovering information in a geographical area, but did not provide an approach for the performance analysis of their proposal. Similarly in [34], the authors have introduced a system called *Locus*. Locus also makes use of a concept similar to FC, but the mechanism to access information is based on queries for specific data items. Moreover, the querying node does not need to be present in the AZ. Therefore, there is a possibility of accessing information without being physically present inside the AZ. This makes Locus different from FC.

In [35] concept called "wireless ad-hoc podcasting" is proposed. It aims at distributing content over an ad-hoc network by making use of a combination of fixed infrastructure (access points) and mobile nodes using ad hoc mode. In the first step each access point fetches content from podcast servers across the Internet and forwards them to mobile nodes within its transmission range. In the second step, a mobile node that has contents to share provides data to another mobile node when they pass within radio range of one another by using opportunistic communication. The difference in concept as compared to FC is that for FC concept, the fixed infrastructure is not present at all and it relies on pure opportunistic communications for spreading of information. Moreover, in the FC concept content is restricted geographically and the content itself has an expiration time referred to as Time to Live, after which content is removed from the network.

In [36], an opportunistic spatio-temporal dissemination system for vehicular networks was proposed, introducing the publish/subscribe communication paradigm to opportunistically disseminate restricted spatio-temporal information in vehicular networks. The subscription information is used to filter and disseminate the events to only the interested nodes. This work exploits specific characteristics of vehicular networks in order to disseminate information more efficiently, and is therefore difficult to generalize to scenarios different from vehicular networks.

In [7], the authors investigated *storage capability*, a concept very similar to FC in VANETs by considering real traffic traces of San Francisco Yellow Cabs. The authors defined *storage lifetime* as the amount of time for which information stays available within the network, and concluded that for one-way highway traffic, transmission range is a key parameter for storage lifetime. Instead, for two-way highway traffic, the size of the geographic region in which we want to disseminate information plays a key role. But in this work the main focus is on the amount of time for which information stays within a geographic region for VANETs, while we look at success probability as a performance indicator.

In [37], the authors proposed a middleware architecture that makes use of access points and opportunistic communications for distribution of content. They also proposed a method to structure content in order to facilitate efficient lookup matching, and a solicitation protocol by using which a node can use content meta-information in order to discover content in its vicinity, allowing it to download content entries from multiple disjoint nodes. They provide an API that can be used to access the system services through a publish/subscribe interface. Their solution is significantly different from what we study in this work, because it uses access points for content distribution. [38] also investigates the benefit accrued from limited infrastructure support for the content having limited geographically scope.

In [38], the authors presented the *Push-and-Track* framework that makes use of 3G networks in order to guarantee short delivery delays for floating data to the new nodes coming in a given geographic region. But for this case, the approach is different from our work, because the focus is on offloading traffic from 3G network to ad hoc network.

In [21], the authors introduced the concept of Floating Content, and derived asymptotic conditions (called criticality conditions) for the expected floating lifetime to be large under some large population assumptions. This corresponds to conditions for the information to remain available in the AZ, and supports the viability of the FC paradigm. The criticality condition depends on three key parameters as shown in (2.1): the average number of nodes in the AZ, the average contact rate experienced by a node, and the average sojourn time of a node within an AZ.

$$\frac{\eta\nu}{\mu} > 1 \tag{2.1}$$

In the fraction above η is the average numbers of nodes in the anchor zone, ν is the average contact rate an ordinary node experiences during its sojourn time in an AZ, and $1/\mu$ is the average sojourn time that a node spends within an AZ. The fraction on the left hand side in the above inequality is the average number of successful encounters a randomly chosen node experiences during its sojourn time. The criticality condition reveals that the average number of encounters for a node during the sojourn time should be greater than 1 in order to ensure the content to float with a high probability. If this value is less than 1 then information is very likely to be lost, and to float for very short time.

In [22], the authors validated the analytical results presented in [21] with extensive simulations, and showed that the criticality condition behaves well under different mobility models and content floats in a reasonably sized AZ even when there are modest number of nodes in the network.

An open issue in these papers is the lack of a correlation between the primary performance parameters from an application perspective and the main design parameters of FC. For concrete applications, it is not sufficient that the content asymptotically floats: for the application performance are vital the density of nodes with content inside the AZ, their spatial distribution, and the percentage of times a node gets the content, once it enters the AZ. Therefore, it is of immense importance to investigate the effect of the system design parameters on the performance of an application using FC. Therefore developing an analytical model for understanding the impact of FC design parameters like AZ radius, node transmission range, and node density on the performance of an application using FC is vital. In addition, evaluating performance under some realistic mobility models is also very important because performance of a communication service such as FC is highly dependent upon the specific mobility patterns of the nodes. Last but not least, performance evaluation by real world experiments by deploying a real application based on FC in different settings can also be very useful for evaluating and understanding the practical feasibility of the FC service for context-aware mobile applications confirming its viability to act as a communication service for context-aware applications.

Chapter 3

A Simple Approximate Analytical Model for Floating Content

In this chapter, we present a simple approximate analytical model for the performance analysis of context-aware applications that use FC. In the previous modeling studies of FC [21, 22] the focus was on understanding the asymptotic properties of the floating lifetime, i.e, the duration of time for which content floats in the AZ. Instead, our objective is to characterize the performance of context-aware applications relying on the FC service. Indeed, from an application perspective, it is not sufficient that the content asymptotically floats: what matters is the probability for a node traversing the AZ to get the content which is floating. In this chapter we address this issue, investigating the effect of the system design parameters on the performance of applications using FC. We focus on the *success probability*, which captures the likelihood for a user to receive the content when traversing the AZ. A key characteristic of our modeling approach is that success probability is computed from few primitive system parameters, most notably the probability density function of the length of the path followed by users within the AZ. This allows our analysis to be generalized to various settings, including different AZ shapes, different user mobility patterns, different user speed distributions, different service and application models. Our main contributions are:

- We develop an approximate analytical model for success probability, with key parameters the AZ radius, the node density, and the node transmission range;
- We apply our model to three different categories of context-aware applications, and derive expressions for their success probability;
- We demonstrate how the model predictions can be used to tune key system parameters to achieve the desired application performance; and

- We validate our analysis with extensive simulations, showing that the predicted success probability is very accurate.

3.1 System Model

We consider mobile nodes in $\mathbb{R}^2$, that form a homogeneous Poisson process with intensity λ nodes per square meter at time $t = 0$.

Mobility Model: We assume that node movements follow the Random Direction mobility model (RDMM) [39], in which nodes independently travel along a straight line, with an angle of movement uniformly distributed between 0 and 2π. We assume that nodes move with a constant velocity v m/s. In [40], it is shown that under this mobility model, the spatial node distribution remains uniform at all time instants. The reason for choosing this mobility model is its simplicity and analytical tractability. We explore the impact of more realistic mobility through simulations in order to determine if our simple model is adequate to capture the first-order effects in Chapter 4.

Content Replication: At any time instant, a node with an information item $\mathcal{I}$ can seed an *anchor zone* (AZ), which is a circular area with radius R meters centered at its current position. Initially, the seeder is the only node with the content $\mathcal{I}$. Every time two nodes *within the AZ* come in transmission range of each other (we call this a contact), if only one of the two nodes possesses $\mathcal{I}$, it communicates $\mathcal{I}$ to the other. We assume there is no supporting infrastructure available, so that nodes must rely exclusively on ad-hoc communication. All nodes are assumed to have the same transmission range of r meters. We assume that $R >> r$, since these are the cases where content floating has practical utility. The probability that information is transferred during a contact is denoted by Q, and takes into account transmission errors, collisions, and contact duration. Note that nodes delete their own copy of $\mathcal{I}$ when they move outside of the AZ.

Performance Metric: In this work, we focus on the scenarios where the node density and the anchor zone radius are large with respect to the transmission radius, resulting in large floating lifetimes on average. In particular, we assume that the criticality condition derived in [21] holds, so that the expected lifetime of content floating is infinite under the fluid limit approximation of [21]. The measure of performance that we use is the probability that a node entering an anchor zone within the floating lifetime, i.e., when at least one node within the anchor zone has content, successfully receives the information item $\mathcal{I}$.

Definition 1 (Success probability). *The success probability $P_s(\tau)$ is the probability that a node receives the content $\mathcal{I}$ associated with an AZ within a time τ after entering the*

Table 3.1: Notations used in the analytical model

Notation	Value
R	Anchor zone radius
$\overline{N}$	Average total number of nodes in anchor zone
$\overline{n}$	Average number of nodes having content in anchor zone
$\overline{m}$	Average number of nodes without content in anchor zone
λ	Node density
L	Average length of the chord in anchor zone
r	Transmission range
v	Node speed (meter per second)
ν	Frequency at which two nodes come within the transmission range r of each other inside anchor zone
Q	Probability of successful transfer of the content

AZ (if it is still in the AZ), or by the time it leaves the AZ (if it leaves it before time τ) conditional on there being at least one node within the AZ with content at the time of the node's entry.

Note that $P_s(\infty)$ is the probability that a node entering the anchor zone within the floating lifetime receives the content before leaving, and will be denoted as P_s. We empirically measure this quantity by simulating a number of AZs and tracking the fraction of nodes that enter within the floating lifetime, and successfully receive the content.

3.2 Analysis

As a first step, we compute an expression for approximating success probability for the general floating content model. Then, we extend our analysis to include two different categories of applications, and derive approximate expressions for success probability for these applications. Table 3.1 summarizes the notations used in our analysis.

3.2.1 General Floating Content

We assume in our derivation that the system is in an equilibrium state (as assumed in [21]), where the average rate of nodes without content $\mathcal{I}$, which get $\mathcal{I}$ inside the AZ, is equal to the average rate at which nodes having $\mathcal{I}$ leave the AZ. In such an equilibrium, the average number of nodes with and without $\mathcal{I}$ within the AZ remains constant. Note that this equilibrium assumption will be a first source of discrepancy with respect to simulations, where we observe periods in which the numbers of nodes with and without content fluctuate (e.g., the number of nodes with content typically grows right after the appearance of a seeder node at the center of the AZ).

Result 1. *Consider an AZ with radius R, node density λ, and nodes with transmission range r and speed v. Let Q denote the probability that two nodes successfully transfer the content $\mathcal{I}$ while they are in contact. Then $P_s(\tau)$ for $\tau \leq 2R/v$ can be approximated as*

$$P_s(\tau) = \int_0^{2R} \frac{l^2}{\pi R^2 \sqrt{4R^2 - l^2}} \cdot$$

$$\cdot \sum_{k=1}^{\infty} \left[1 - \left(1 - \frac{Q\overline{n}}{(\overline{m} + \overline{n})} \right)^k \right] \frac{(2r\lambda(l \wedge v\tau))^k e^{-2r\lambda(l \wedge v\tau)}}{k!} dl \tag{3.1}$$

where $\overline{m} = \min(\frac{v}{Q\nu R}, \lambda\pi R^2)$, $\overline{n} = \lambda\pi R^2 - \overline{m}$, with ν given by $\frac{2rv^2}{(\pi R^2)}$.

Here, $a \wedge b$ stands for $\min(a, b)$. $\overline{n}$ and $\overline{m}$ are respectively the average number of nodes with and without content within the anchor zone. The derivation of this result is presented in the Appendix A.

Note that in the expression above, the integral is over l which is the length of the AZ chord traversed by a node. The expression calculates the probability that a node meets k other nodes during its traversal as the product of the pdf of the chord length and the conditional pdf of the number of contacts given the chord length. The term in square brackets is the probability that at least one out of the k nodes met has the content, and that the transfer is successful. In deriving the above result, we use a similar assumption as used in [21, 41] that the distribution of nodes with content in the AZ is uniform, and that the odds of meeting nodes possessing the information are uncorrelated. In reality, both assumptions are not satisfied, since there is some spatial clustering of nodes with content, and also a higher density of such nodes near the center of the AZ. However, as we show through simulations in the sequel, these cause second-order effects and the result in (3.1) captures the success probability well.

For some applications that use FC as communication service, more important than computing the probability that a node obtains content $\mathcal{I}$ within a time τ after entering the AZ, is estimating the probability that a node obtains the content before leaving the AZ. We denote this probability as P_s.

Corollary 1. *With the same assumptions as in Result 1, the success probability P_s can be approximated as $P_s(2R/v)$.*

In the derivation of Result 1, this corresponds to integrating over the whole length of the chord traveled within the AZ.

In these results, all factors influencing the probability of successful content transfer between two nodes in range of each other are captured by the parameter Q. This allows detailed models for information transfer at physical and/or MAC layer to be easily incorporated in the analysis, without changing the structure of the formula. In the following theorem we provide a simple expression for the probability of successful content transfer, which accounts for finite bandwidth availability and transmission errors, assuming that nodes continually retry on failure as long as they stay within transmission range.

Theorem 1. *The probability of successful transfer of content between two nodes, assuming that the minimum required time for the transfer is X', is given by*

$$Q(X') = \sum_{k=1}^{\infty} \int_{kX'}^{(k+1)X'} [1 - (1 - S)^k] f_\tau(t) dt \tag{3.2}$$

where $f_\tau(t)$ is the probability density function of the contact duration under the RD mobility model, given by

$$f_\tau(t) = \int_0^{min(2v, \frac{2r}{t})} \frac{2\omega^3 t^2}{\pi^2 r^2 \sqrt{4r^2 - \omega^2 t^2} \sqrt{4v^2 - \omega^2}} d\omega \tag{3.3}$$

and S is the probability of no transmission failures (errors, collisions, etc) for each content transfer attempt.

The derivation of this result is presented in Appendix A.

3.2.2 Application-specific analysis

The performance parameter "success probability" which we considered so far is relevant when FC is used to ensure that users traversing a given AZ (we assumed it to be circular, but extensions to different shapes are simple) get the associated content. However, in order to ensure acceptable application performance, content could be floated in a geographic area that is a superset of the area where it is needed. Therefore, we consider an AZ with radius R_2 which acts as the "replication range" within which content is replicated using FC. A new zone with radius R_1, with $R_1 \leq R_2$, called Range Of Interest (ROI) is also defined. ROI is application-specific and depends on the particular service which has to be delivered. Note that the absolute and relative values of R_1 and R_2 can be chosen so as to achieve the desired system (rather, application) performance. Below, we consider two application categories that have different interpretations for the successful delivery of content.

Application category 1: The expression for success probability for application category 1 is calculated in Result 2 using a new definition for success probability. For more details about applications belonging to this category, please refer to Section 2.1.

Definition 2. *The success probability for getting content $\mathcal{I}$ before leaving the ROI (P_{SBL}) is the probability that a node entering the ROI gets the content $\mathcal{I}$ before exiting the ROI, conditional on the presence of at least a single node with content in the AZ at the time of the node's entry.*

Result 2. *For an AZ with radius R_2 and a ROI with radius R_1, with $R_1 \leq R_2$, P_{SBL} can be approximated as*

$$P_{SBL} = \sum_{k=1}^{\infty} \left(\int_{\sqrt{R_2^2 - R_1^2}}^{R_1 + R_2} \left(f_{L_1}(\ell) \frac{(2r\ell\lambda)^k e^{-2r\ell\lambda}}{k!} \right) d\ell \right) \left[1 - \left(1 - \frac{Q\overline{n}}{(\overline{m} + \overline{n})} \right)^k \right] \tag{3.4}$$

where $f_{L_1}(l)$ is given by

$$f_{L_1}(l) = \frac{2\sqrt{R_2^2 - (g^{-1}(l))^2}\sqrt{R_2^2 - (g^{-1}(l))}}{(\int_0^{R_1} g(y)dy)g^{-1}(l)} \tag{3.5}$$

and $g(y) = \sqrt{R_2^2 - y^2} + \sqrt{R_1^2 - y^2}$.

The proof of Result 2 is presented in Appendix A.

Application category 2: For a different type of application, it is important to deliver the content $\mathcal{I}$ to users before they enter a certain area (which we identify again with the ROI). The expression for success probability for such kind of applications is derived in Result 3, using yet another definition for success probability. For more details about applications belonging to this category, please refer to Section 2.1.

Definition 3. *The success probability for getting content $\mathcal{I}$ before entering the ROI (P_{SBE}) is the probability that a node entering the AZ gets the content $\mathcal{I}$ before entering the ROI, conditional on the presence of at least a single node with content in the AZ at the time of the node's entry.*

Result 3. *For an AZ with radius R_2, and a ROI with radius R_1, with $R_1 \leq R_2$, P_{SBE} can be approximated as*

$$P_{SBE} = \sum_{k=1}^{\infty} \int_{R_2-R_1}^{\sqrt{R_2^2-R_1^2}} \left(f_{L_2}(\ell) \frac{(2r\ell\lambda)^k e^{-2r\ell\lambda}}{k!} \right) d\ell \left[1 - \left(1 - \frac{Q\overline{n}}{(\overline{m}+\overline{n})} \right)^k \right] \tag{3.6}$$

where $f_{L_2}(l)$ is given by

$$f_{L_2}(l) = \frac{2\sqrt{R_2^2 - (h^{-1}(l))^2}\sqrt{R_2^2 - (h^{-1}(l))}}{(\int_0^{R_1} h(y)dy)h^{-1}(l)} \tag{3.7}$$

and $h(y) = \sqrt{R_2^2 - y^2} - \sqrt{R_1^2 - y^2}$.

The proof of Result 3 is presented in the Appendix A.

3.3 Simulations and Results

In this section, we present and discuss numerical results obtained from our approximate models of the previous sections, and from simulations. For all simulation experiments, the OMNeT++ based framework called INET [42] is used. Confidence intervals at 95% confidence level were evaluated for all cases using independent replications, and are shown in the following figures, with the exception of Fig. 3.2, in order to avoid cluttering the graph. The purpose of the presentation of numerical results is twofold. On the one hand, we validate the approximate expressions derived for success probability, showing that they are accurate under varied conditions. On the other hand, we show the effectiveness of the analysis in selecting the FC parameters for different applications and under different scenarios. In simulations, the average user density ranges from 22 to 66 nodes per square kilometer. The transmission range is 50 meters and nodes move with a constant speed of 10 meters per second. Anchor zone radius ranges from 500 to 1000 meters. We simulate multiple instances of anchor zones, and measure success probability in each instance over the floating lifetime or until a maximum of 50000 seconds have elapsed. Table 3.2 summarizes the simulation parameters used in our study.

Table 3.2: Simulation parameters

Simulation Parameter	Value
Average user density	$22 - 66$ nodes per square kilometer
Node speed	10 meters per second
Anchor zone radius	$500 - 1000$ meters
Mobility model	Random Direction Mobility
Transmission range	50 meters
Simulation time	50000 seconds

3.3.1 Results for General Floating Content

Fig. 3.1 shows both the analytical predictions and the empirically determined values of success probability as a function of the AZ radius.

It can be seen that the model predictions match very well with the simulated results, suggesting that the model indeed captures successfully the first order-effects on success probability. The curves in the figure also show that an increase in both node density and AZ radius improve the success probability. An increase in the the AZ radius increases the average time a node spends inside the AZ, therefore the chances of meeting a node having content $\mathcal{I}$ increase. Similarly, an increase in node density also results in higher overall contact rate as well as a higher chance of meeting a node with content $\mathcal{I}$. Fig. 3.1 also shows the impact of transmission errors and finite bandwidth. It can be seen that under a finite bandwidth model with a data rate of 11 Mbps, for transmission error probability of 0.2 with a file size of 2MB, the success probability decreases. As the transmit errors and limited contact times reduce the rate of contacts where communication is successful, this reduces the fraction of nodes in the AZ with content as well as the overall success probability.

Fig. 3.2 shows success probability versus the node density for different choices of AZ radius. As the node density increases, a large improvement in success probability can be observed. The analytical model captures this effect, and is a very good predictor for success probability. Further, Fig. 3.2 shows that as AZs grow larger, a given success probability threshold can be achieved at lower node densities. This is due to node paths through the AZ getting larger with AZ size, resulting in more opportunities to obtain content. However, if node densities and AZ size are varied jointly such that the average number of nodes in the AZ stays unchanged (see the isoline corresponding to an average of 65 nodes in the AZ), larger AZs result in lower success probabilities. The key parameter behind this effect is the ratio between the AZ radius (R) and the node transmission

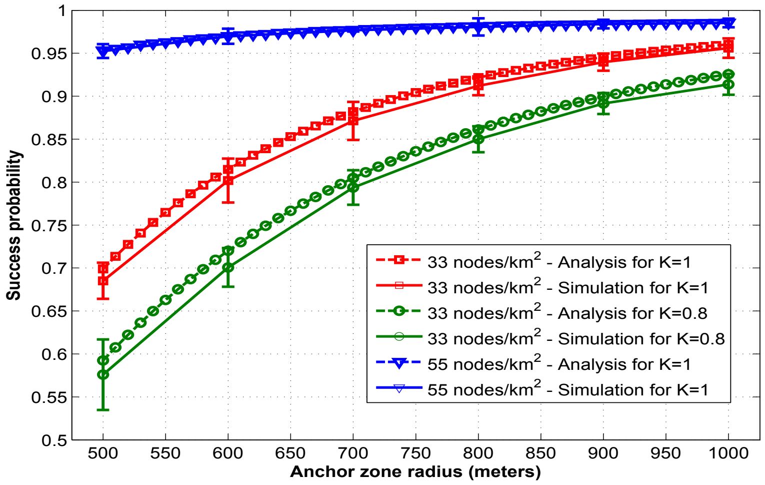


Figure 3.1: Success probability vs. AZ radius.

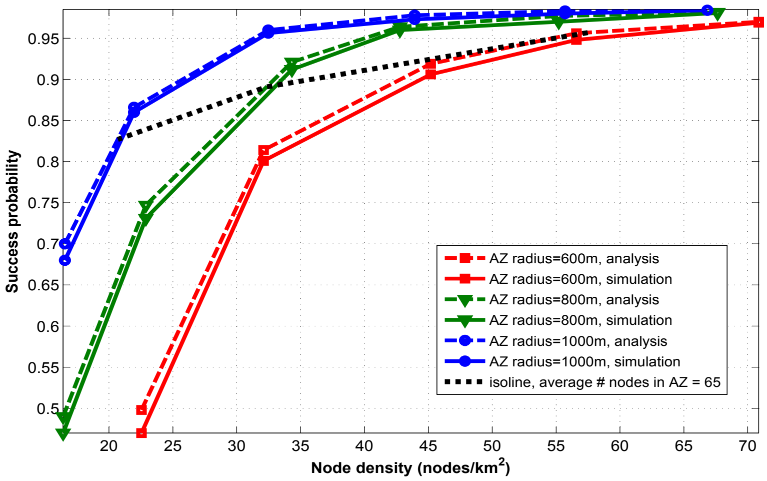


Figure 3.2: Success probability vs. node density.

range (r). Indeed, as the AZ radius increases, this ratio decreases, and as a result more nodes are needed in the AZ in order to achieve a similar success probability. Clearly, defining very large AZ might lead to wastage of resources without significantly improving performance. Thus, the ratio of the AZ radius to the transmission range is a critical parameter, that must be tuned carefully.

3.3.2 Results for Applications

Figs. 3.3 and 3.4 show curves of success probability versus the AZ radius R_2, for different node densities, for application categories 1 and 2, when ROI radius (R_1) is 200m. Increases in either AZ radius (R_2), or node density result in increased success probability

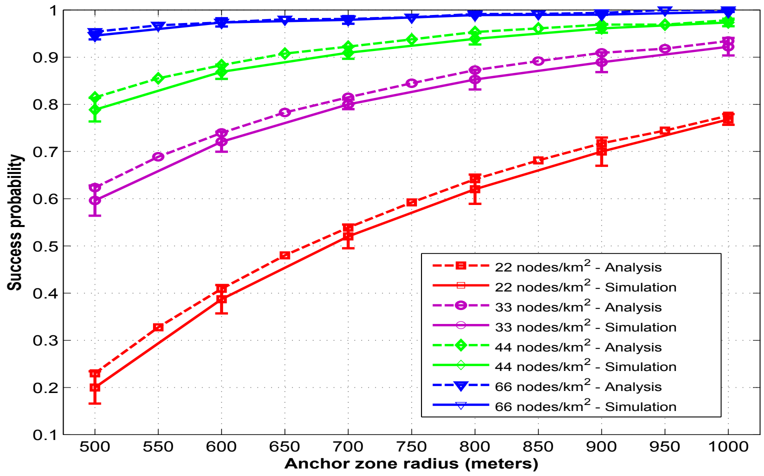


Figure 3.3: Success probability for application 1 with ROI = 200m.

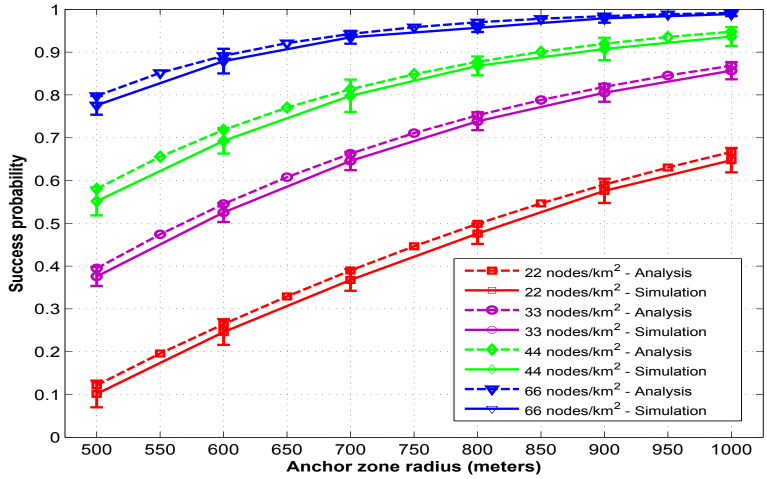


Figure 3.4: Success probability for application 2 with ROI = 200m.

for both applications. In both cases, nodes traversing the AZ make more contacts resulting in a larger fraction of nodes having content in the AZ, and more opportunities for a node to obtain the content. We see that increasing AZ radius has diminishing returns, with this effect being more pronounced in the case of application 1. It can be seen from both figures that our analytical results are very close to simulations results.

In a real-world environment, an application using FC is likely to require a minimum success probability (a shop may require that a given percentage of the people passing within 200 m of its premises receive their advertisement). However, unnecessarily large AZs would lead to resource wastage. From Figs. 3.3 and 3.4, we see that the proposed analytical model can be used in order to tune AZ radius to achieve the desired success probability. Here, we assume the threshold to be 90%, and use our models to compute

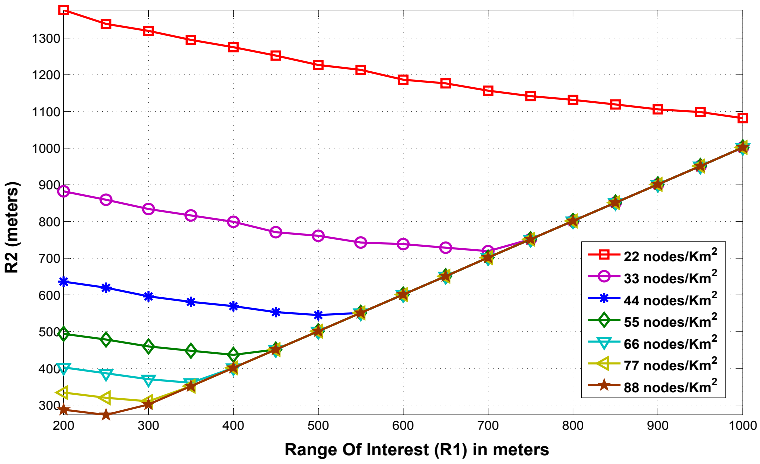


Figure 3.5: AZ radius to achieve 90% success probability for application 1.

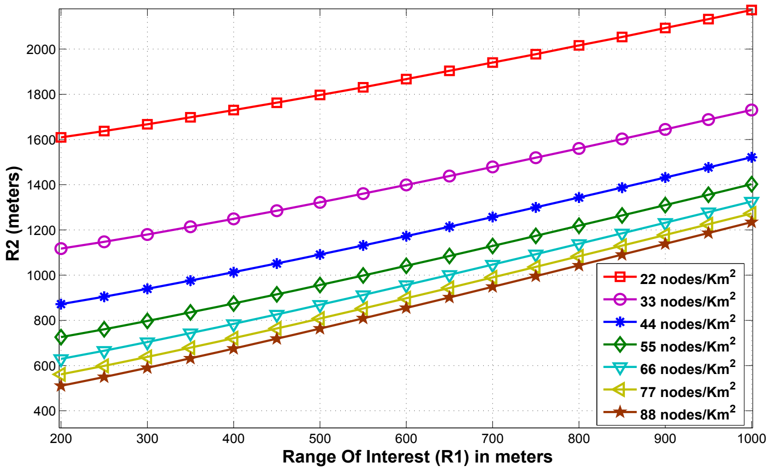


Figure 3.6: AZ radius to achieve 90% success probability for application 2.

the minimum AZ radius (R_2) that is required to achieve this objective at different ROIs and node densities. Figs. 3.5 and 3.6 depict the results.

For application 1, in Fig. 3.5 we can observe that, for a given node density, as ROI increases, the required R_2 decreases until the condition is reached where replicating the content within the ROI is sufficient to achieve the desired success probability. When such condition is reached, R_1 becomes equal to R_2 (we always consider $R_1 \leq R_2$). In the case of application 2, we observe from Fig. 3.6 that, for a given density of nodes, the required AZ size (R_2) increases as R_1 increases. Here, as the ROI increases, we also need larger R_1 in order to make sure that node paths in the AZ are long enough for them to learn the content with sufficiently high probability before entering the ROI.

3.4 Summary

In this chapter, we focused on FC and its ability to act as the communication paradigm supporting context-aware applications. We defined success probability as the primary performance indicator, and developed a simple, approximate analytical modeling framework, which can be adapted to several different settings. Indeed, our models essentially depend on the length of the path within the AZ where content floats and the spatial density of nodes. Different settings (in terms of AZ shape, user speed distribution, application or service characteristics, etc.) may correspond to different path lengths, but the analytical approach we presented in this chapter remains applicable. We showed by simulation that our approximation computes very accurate success probability values for a wide range of anchor zone radii and node densities.

Our models can be adapted to several different categories of context-aware applications, and the model predictions can be used in order to tune key parameters of the system to achieve the required performance with minimum overhead. We studied two such cases in this chapter, deriving approximate expressions for success probability for both. In addition, in these two cases, we validated our model using extensive simulations in OMNeT++, proving the very good accuracy of our analytical predictions. Our simulation results showed that high success probabilities are achieved with reasonably sized anchor zones that are only slightly larger than the region of interest even at low node densities, demonstrating the viability of FC as an enabler for context-aware applications.

Chapter 4

Impact of Different Mobility Models on the System Performance

In this chapter we extend the work presented in previous chapter and investigate the impact of different mobility models on the performance of context-aware applications using FC. Random Direction Mobility Model (RDMM) is a very simple mobility model and does not capture the complexity of realistic movement patterns. Hence, it is important to evaluate the FC performance under different and more realistic mobility settings. Therefore, in addition to RDMM, we also simulate Reference Point Group Mobility (RPGM), to account for group mobility [43], Manhattan Grid Mobility Model (MGMM), which provides a simplistic model for vehicular mobility [44], and a synthetic mobility trace based on real vehicular traffic statistics collected in the frame of the TAPASCOLOGNE project [45] in the city of Cologne, Germany.

With our experiments, we want to verify by simulation how well FC behaves in realistic mobility settings, and how closely the values of success probability predicted by our simple analytical model match those obtained with complex mobility models.

4.1 Mobility scenarios

One of the most important aspects impacting the performance of the FC service is the way in which users move in space. We investigate this aspect, considering three different mobility models and a set of realistic vehicular traffic traces, and assessing the relationship between their characteristics and the performance of FC through extensive simulation experiments.

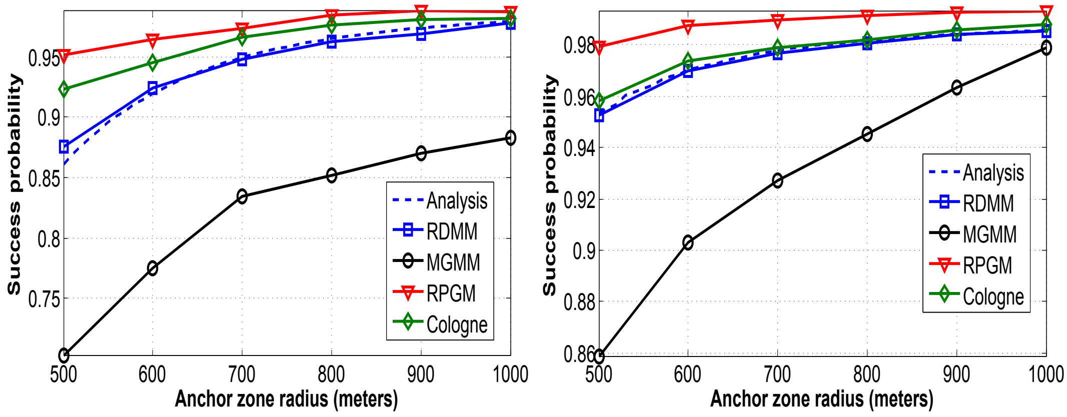
The first mobility model we consider is the Random Direction Mobility Model (RDMM), one of the most commonly used, and the one underlying the derivation of Result 1.

In RDMM, nodes independently travel along a straight line, with an angle of movement uniformly distributed between 0 and 2π. This mobility model is simple and easily tractable analytically because the spatial node distribution remains uniform at all time instants [39].

The second model is the Manhattan Grid Mobility Model (MGMM), used to describe the mobility of vehicles in an urban area [44]. It uses a grid road topology for modeling the movements of vehicles. At each road junction, each vehicle may turn left, turn right or continue straight according to some given probability, which can be tuned to obtain different mobility behaviors. We chose it in order to analyze the impact of a grid topology, typical of a modern city, on the performance of FC when used by applications residing on vehicles.

The third model is the Reference Point Group Mobility model (RPGM), a group mobility model [43]. We have chosen it to evaluate the impact on the performance of FC of clustering and of correlation in user mobility patterns. In RPGM, nodes move in the form of a group and each group has a geographical scope. Nodes belonging to a group are uniformly distributed within its geographical scope. Each group has a *logical center* and all the nodes belonging to that group follow the *logical center*. This logical center moves according to a group motion vector $\vec{V_g}$. For individual movement of nodes, each node is assigned a reference point which follows the group motion vector. After time τ, a new reference point is calculated by adding a random motion vector $\vec{RM}$ to the group motion vector $\vec{V_g}$. The length of $\vec{RM}$ is uniformly distributed within a certain radius centered at the reference point, and the direction is uniformly distributed between 0 and 2π. Adding a random motion vector enables a random motion behavior for each individual node. Different mobility scenarios can be modeled with RPGM. One example is groups of tourists visiting some famous attractions in a city. Another example is mobility in a disaster recovery area where different medical assistant teams, rescue teams, firemen teams are randomly moving in the area for the help and rescue operation.

For the fourth considered scenario, we use synthetic mobility traces from the city of Cologne. The Cologne dataset is one of the largest freely available realistic traces capturing both macroscopic and microscopic features [45]. It is realistic from a microscopic point of view because it captures the realistic movement of individual drivers in presence of other vehicles, traffic signals, road junctions, etc. From a macroscopic point of view, it mimics the evolution of large traffic flows across a metropolitan area over time.



(a) 44 nodes/km^2

(b) 66 nodes/km^2

Figure 4.1: Success probability for baseline application.

4.2 Performance evaluation

For all simulation experiments, we use the OMNeT++ based framework called INET [42]. We evaluate the performance under the four different mobility scenarios previously described. We also compare the simulation results with the analytical results obtained through our simple analytical model presented in Chapter 3. When considering RDMM, MGMM and RPGM, each node moves with a constant speed of 5 m/s, while in the Cologne dataset vehicles move at variable speed. For the case of MGMM, a block size of 200 m $\times$ 150 m is used. We simulate various values for the AZ radius, while we keep the ROI constant and equal to a circle or radius 200 m at the center of the AZ. As the Cologne dataset is very large, covering a region of 400 Km^2, for our simulations we considered an area of 9 Km^2 at the center of the city, and a two-hour time interval (from 6 AM to 8 AM) [46]. For the MGMM, the probability of turning right, turning left and going straight are, respectively, 0.25, 0.25 and 0.5, mimicking typical behavior of cars in city centers. The end-user transmission range is always taken to be 50 meters.

Fig. 4.1 shows values of success probability as a function of the AZ radius for all the considered mobility models for the baseline application, for two values of nodes density. It can be seen that an increase in either the AZ radius or the node density improves success probability for all scenarios. The reason behind this behavior is that a larger AZ radius increases the average time a node spends inside the AZ, while a higher node density increases the contact rate, both resulting in more chances of meeting a node having content, and thus in higher success probability. For a given AZ radius and node density, RPGM yields the highest success probability, showing that clustering has a beneficial impact on the propagation of the content within the AZ and on its availability. The success probability predicted by our analytical model is very close to the ones by simulations for RDMM, for which the model was developed. We note that MGMM yields a lower success probability than RDMM in all cases. There are two main reasons behind this. First of all, if we look at the path length distribution within the AZ for MGMM and RDMM (see Fig. 4.2), we see that, unlike in RDMM, in MGMM shorter path lengths have a high probability as compared to relatively longer ones. For the considered block size, a high percentage of nodes traverse shorter paths inside AZ, which reduces the probability of meeting a node with content. The second reason is that, assuming that block size is much larger with respect to the node transmission range, and nodes move with a constant speed, a node can meet another node only if the other node is moving in the opposite direction (if both of them are on same road) or at the road junctions (where a node can meet other nodes traveling in other directions). This reduces the contact rate, resulting in decreased chances of meeting a node with content. MGMM and Cologne mobility traces are somewhat similar, in what they are both based on a grid of streets in a urban area. However, unlike MGMM, in a realistic setting like the one in the Cologne dataset, vehicles stop at intersections due to traffic signals, and also move according to car following model, which represents a realistic driver behavior [45]. Moreover, nodes move with variable speed, unlike in MGMM, where speed is constant. This also results in increased contact rate, and larger probability of meeting a node with content, resulting in increased success probability in the case of Cologne mobility. Moreover, MGMM keeps nodes uniformly distributed on all the area, while we have verified that mobility patterns in the Cologne dataset exhibit some correlation between vehicles mobility patterns, and some degree of clustering (traffic jams, traffic lights, etc), which, as it happens for RPMM, improve the performance of FC.

Continuing the comparison between the results for MGMM and RDMM, an interesting observation is that, in case of application categories 1 and 2, the path length of users entering the ROI cannot be shorter than the difference between the AZ radius and the ROI radius. Therefore, we can expect that the success probabilities for application categories 1 and 2 are not impaired by the path length distribution shown for MGMM. Indeed, as shown in Fig. 4.3 and Fig. 4.4, the success probability of application categories 1 and 2 under MGMM is closer to that of RDMM, as compared to the baseline case. Under the topological settings used for the experiment reported in the figure, the minimum path length for application categories 1 and 2 is 300 m. This means that for application categories 1 and 2, the success probability is computed for paths inside the AZ with length greater than 300 m. This leads to considering longer paths as compared to the baseline case, and as a result the success probability increases and approaches the one given by

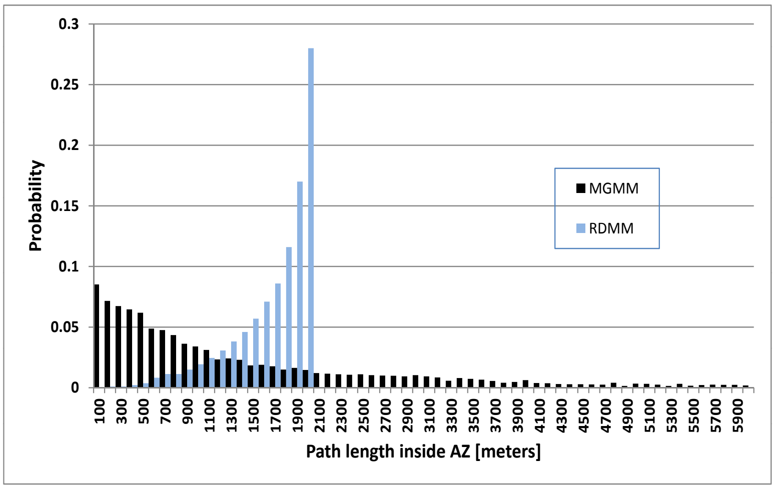


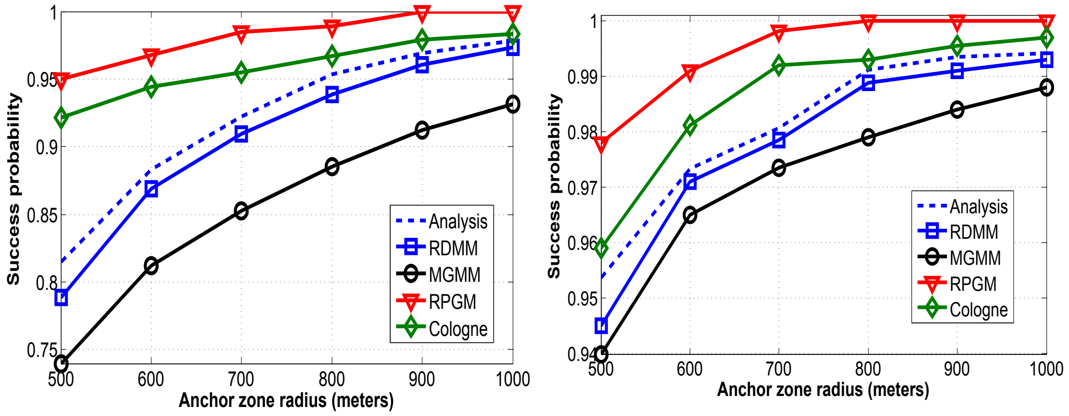
Figure 4.2: Distribution of path lengths inside AZ for MGMM 1000 m AZ radius.

the simple analytical model.

Specifically, Fig. 4.3 shows curves for success probability versus the AZ radius, for application category 1 under node densities of 44 and 66 $nodes/km^2$ respectively. For the Cologne traces, the plots have been obtained by individuating two time intervals, of 1000 seconds each, during which the average node density in the considered area is equal to the values of node density previously mentioned. As expected, increasing the AZ radius results in higher success probability for application category 1, under all mobility models. The reason is that increasing the AZ radius results in longer average amounts of time a node spends in AZ, resulting in increased chances of meeting a node with content. For all the considered mobility models, our analytical model predictions of success probability become more accurate for higher node densities. This is due to the assumptions underlying its derivation, which hold for a large number of nodes in the AZ.

Fig. 4.4 shows curves for the success probability versus the AZ radius, for application category 2 under node densities of 44 and 66 $nodes/km^2$ respectively. The behavior is similar to Fig. 4.3. For all the considered mobility models, larger AZ radiuses translate in increased success probability for application category 2. If we consider an accident warning application, where the objective is to notify the nodes entering an area close to the accident location, so that a driver can make an informed decision, we can observe from Fig. 4.4 that FC is capable of providing a reasonably high success probability.

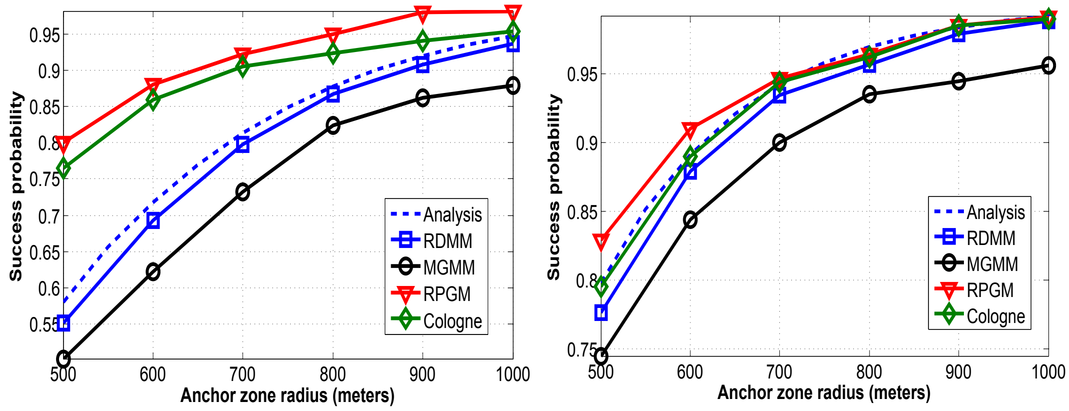
From our evaluation, we can conclude that FC is a very useful communication paradigm that can be used for a variety of context-aware applications. If parameters are carefully tuned/configured, it is capable of providing a reasonable success probability for a variety of applications and of user mobility patterns. The success probability values predicted by our simple analytical model are quite close to the values obtained from simulations for



(a) 44 nodes/km^2

(b) 66 nodes/km^2

Figure 4.3: Success probability for application category 1.



(a) 44 nodes/km^2

(b) 66 nodes/km^2

Figure 4.4: Success probability for application category 2.

the case of RDMM. A better representation of the path lengths within the AZ is necessary to obtain comparable accuracy for the other considered mobility models, especially MGMM.

4.3 Summary

In this chapter, we focused on the impact of end-user mobility models on the performance of context-aware applications using FC as a communication paradigm. We considered three different categories of context-aware applications, and four different user mobility models. We found that FC can provide very effective performance to a variety of context-aware applications under quite diverse mobility patterns. Comparing simulation results to the performance predictions of a simple analytical model that was developed for RDMM, we found a very good agreement in the case of RDMM, as already observed in [24]. Other mobility models call for some model re-working to achieve similarly accurate estimates. For all the considered mobility models, high success probabilities can be achieved by tuning the anchor zone radii for a variety of context-aware applications, which justifies the viability of FC as an enabler for context-aware applications.

Chapter 5

Experimenting with Floating Content in an Office Setting

In this chapter, we explain the experimental performance evaluation of FC in an office environment. Previous studies of the performance of the FC service focused on determining the conditions under which the content floats with very high probability [21, 23] and on application-level performance modeling [24, 47]. All these works are based on simplifying assumptions for user mobility, and on a simplified model for data exchange. The resulting models give indications about the potential and limitations of FC, but they are only first-order predictions of performance in realistic settings. Indeed, important aspects, such as the combined impact of the actual mobility patterns of users, of the communication protocols, of the specific propagation characteristics in the chosen area, of localization accuracy, are difficult to investigate analytically and to evaluate by simulation. Moreover, it is important to understand the impact of the limitations and of the specific features of available protocols for opportunistic communications (such as Bluetooth and WiFi Direct) on the performance of the FC service. Therefore, in this chapter we present the first evaluation of the impact of all those factors on FC performance and discuss the practical feasibility of the FC service for context-aware mobile applications in an office setting.

5.1 Experimental Evaluation of FC

In this section we describe our implementation of the FC service on a smartphone application, and we present our test scenarios for the study of the performance of our *FC mobile app*. The experimental setting we chose is an office (the IMDEA Networks Institute main site, namely INI), composed by a mix of open spaces, meeting rooms and classic, one-room offices (see Fig. 5.1). For all tests, we chose a ROI coinciding with the AZ in order to simplify the implementation and the assessment of the FC service.

5.1.1 Floaty: a Floating Content Mobile App for Android

In order to perform our tests, we implemented the floating content service on an Android smartphone application, namely the *Floaty app* [48]. We chose to use Android because of its wide diffusion among existing mobile users. For the exchange of messages among smartphones, we chose to employ Bluetooth, as it is available in almost all the smartphones and it typically consumes much less energy than WiFi [49]. We also verified through simulations that, for a range of node densities which is typical of office settings, and under a variety of mobility models, the Bluetooth transmission range is sufficient to achieve more than 90% success probability. Indeed, the only other technology currently available on smartphones for ad hoc communication is WiFi Direct. However, Bluetooth is currently preferable to WiFi Direct because the latter is only supported by devices running Android version 4.0 or above (although some Android 2.3 devices through proprietary operating system extensions developed by OEMs have this feature). Moreover, currently WiFi Direct poses several technical problems due to the lack of user transparent authentication modes. Finally, the dimensions of the area chosen for the measurements is such that using WiFi Direct for opportunistic message transfers would have created a full mesh network within the anchor zone, preventing us from analyzing the effects of user mobility patterns on FC performance.

Whenever a smartphone running the Floaty app is in the AZ, it generates a new message every T minutes, which is transferred to all hosts in range running the same app. In our experiments we chose T equal to 15 minutes. In order to univocally identify it, every floating content is tagged with an identifier of the phone which generated it (its MAC address and Bluetooth name), with the time at which it has been generated, and with a sequence number.

Every D seconds, Floaty retrieves the list of available Bluetooth peers. For each peer, it checks whether it is running the app as well (the presence of the application running on the device is determined by a Universally Unique Identifier (UUID), broadcasted within the scan response data). Then the Floaty app chooses a peer among those running the app, and it tries to establish a connection to it. If it succeeds, each of the two peers sends all the messages it is currently storing. Each peer then retains only the messages it did not already have. After this, the connection is closed. Due to Bluetooth limitations, Floaty cannot handle simultaneous connections. In our experiments we chose D equal to 1 minute. Since scanning and connecting/disconnecting from a Bluetooth peer takes several seconds, it is important to optimize the choice of the peers, to improve content diffusion in dynamic settings. To this end, the application maintains a list of content exchanges, and prioritizes connections to those peers to which it did not send a given content before. We assumed the AZ to be defined as the area covered by the signal of at least one of a set of reference WiFi access points (APs). In order to understand whether it is in the AZ or not, every P seconds each app retrieves the list of available APs, and it checks the presence of at least one reference AP. To mitigate the effects of instantaneous fluctuations in APs signal intensity on the localization process, the application makes use of a temporal hysteresis. Whenever it finds no reference APs in n consecutive scans, the app assumes to be out of the AZ, and it deletes all stored messages. Similarly, the app considers to be in the AZ whenever it finds at least one reference AP in m consecutive scans. In our tests we verified experimentally that the values $n = m = 2$ and $P = 30$s are sufficient to avoid localization flickering at AZ borders, and to strike a good balance between localization accuracy on one side, and the delay with which an ingress/egress into/from the AZ is registered by the app on the other. The resulting AZ is generally not circular, and it may have holes due to shadow zones.

In order to measure the performance of the floating content service, the application logs all events related to message transfer between hosts, associating them with a timestamp, with a duration of the transfer, and with the list of transferred messages. The application also logs all events relating to AZ ingress and egress, as well as all failed message transfers. These logs are periodically uploaded to a server: whenever the application has some logs to send, it regularly checks for connectivity to the Internet (via WiFi or 3G) and, whenever available, it sends the results to the server.
The application does not require interaction with the smartphone's owner and it runs in the background. It is based on Android native API and is compatible with all Android terminals with Android version of 2.3 and later. For more details on the app and its implementation, please refer to [17].

5.1.2 Impact on utilization of system resources

In a first set of tests, we characterized our app in terms of system resources utilization. Our initial goal indeed was to assess the memory and CPU utilization, as well as energy consumption (and therefore impact on the smartphone autonomy) of our app, while content is being generated and replicated. We installed the Floaty app on 12 smartphones of different brands and with different Android versions. The AZ in this case covered the entire INI premises at the second floor of an office building (Anchor Zone 1 in Fig. 5.1), for a total area of 800 m^2. This AZ coincided with the area served by all of INI's WiFi

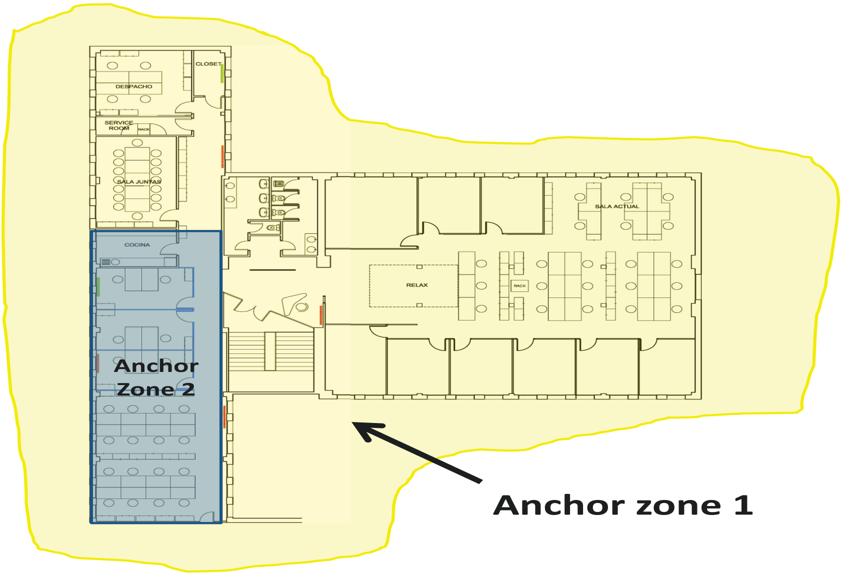


Figure 5.1: Planimetry of the site of the experiments.

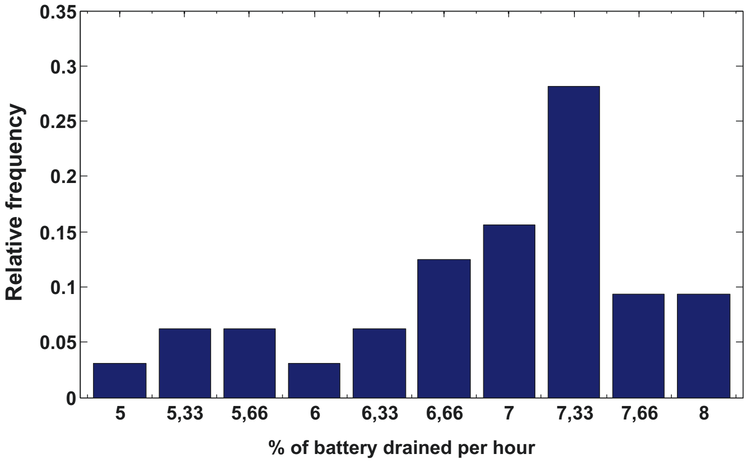


Figure 5.2: Battery discharge per hour due to Floaty app.

access points. Each phone running the Floaty app generated a new message every 15 min, and every minute each instance of the Floaty app searched for Bluetooth hosts running the same app, with which messages could be exchanged. In order to maximize the reproducibility of our tests, no application was active on smartphones other than Floaty, and all screens were kept off. Every test lasted 4 hours. As a baseline, we also measured resources utilization on the same pool of smartphones when the Floaty application is turned off.

We first analyzed the increase in battery consumption due to the Floaty app, while

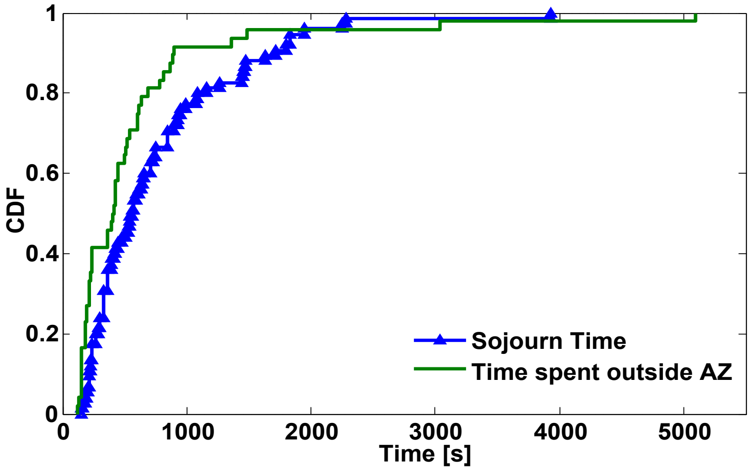


Figure 5.3: CDF of sojourn time, and of the time spent outside of the anchor zone.

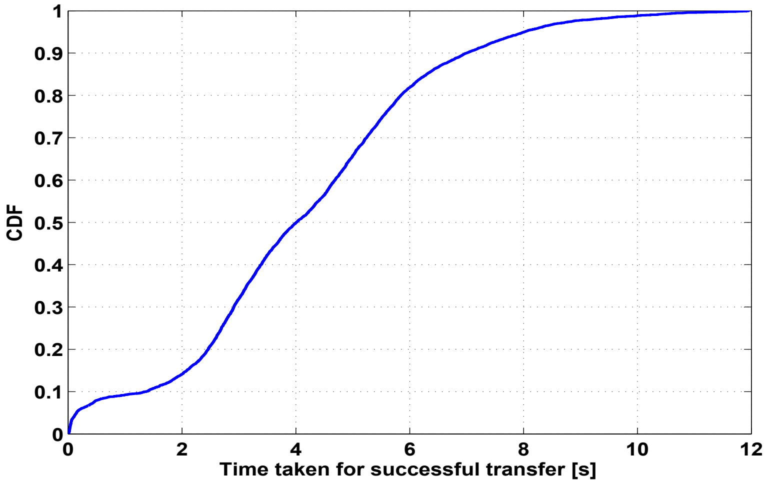


Figure 5.4: CDF of the time required for successful transfer.

content was generated and exchanged in the AZ. We found that, over all phones, the decrease in battery charge due only to the Floaty app was between 5% and 8% per hour (as shown in Fig. 5.2), with an additional 1% consumed by the OS (i.e., the same smartphones, running no app, consume around 1% per hour). Over the same set of tests, we verified that the average total smartphone CPU utilization never goes above 45% and that the memory used by the app never goes above 46 MB. We observed that both these quantities grow with increasing number of message transfers, as it may happen, for instance in settings with higher density of users. Overall, this initial set of tests allowed us to verify that the Floaty app implies a resource utilization level which does not impair the global performance of the smartphone or substantially decrease its battery lifetime.

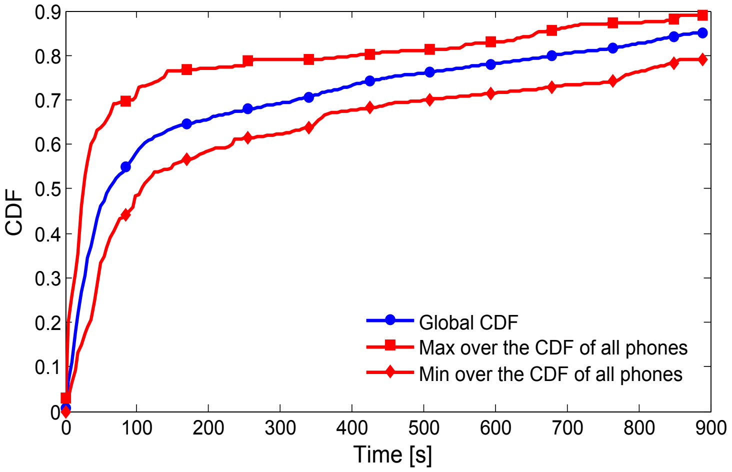


Figure 5.5: CDF of the time to get the content (values higher that $900s$ and undelivered contents are not represented in the figure).

5.2 Performance assessment of the FC app

In this section we report the main results of our experiments. For performance evaluation, we use the following definitions for content availability and success ratio.

Definition 4 (Availability)**.** *The availability of a content after t time units from its generation is the fraction of users inside the AZ holding a copy of the content.*

Definition 5 (Success ratio)**.** *The success ratio of a content is the ratio between the number of nodes that received the content during the content lifetime and the number of nodes that had the opportunity to obtain the content either because they were present in the AZ at content generation time or because they entered the AZ during the content lifetime.*

In order to experimentally assess the performance of the FC service in an office setting, we ran the Floaty app on 8 smartphones, and the AZ covered only a portion of the INI premises (Anchor Zone 2 in Fig. 5.1). The AZ coincided with the area served by just one of INI's WiFi access points. The choice of a smaller AZ with respect to the previous set of tests was made to allow frequent ingresses and egresses of users from the AZ during the experiment. Whenever the app is in the AZ, it generates a new content every 15 min, and every minute each instance of the Floaty app searches for Bluetooth hosts running the same app, with which contents can be exchanged. Conversely, when the app is running outside of the AZ, it does not generate floating contents nor it replicates any content.

Every test lasted 4 hours during office working hours, so that most of the smartphone owners are within the INI premises. As expected, in the chosen application context, users

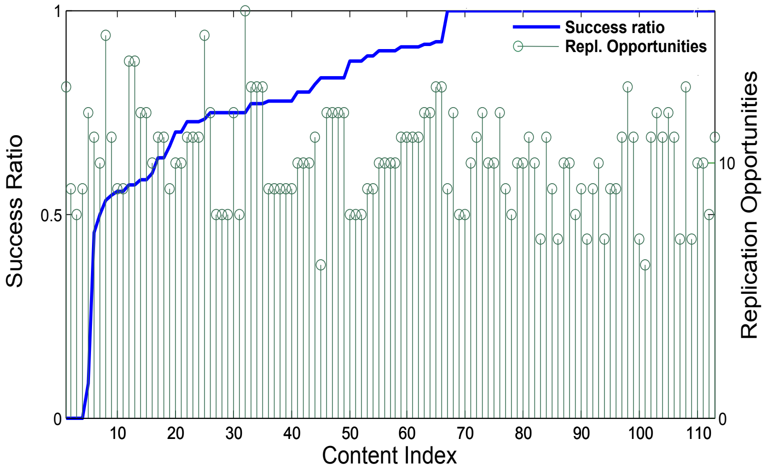


Figure 5.6: Content success ratio and number of entries in AZ for each content.

exhibit the typical mobility pattern of office employees, spending long periods of time at their desk, but frequently walking within the premises for meetings, to interact with colleagues, as well as for coffee and lunch breaks, and always carrying their smartphone with them. Several of the owners of the smartphones used in our tests have their seat in the area of the AZ. In order to characterize the mobility pattern of users, in Fig. 5.3 we plotted the CDF of the smartphone sojourn time within the AZ, as well as of the time spent outside of the anchor zone. We can see how the average amount of time spent in the AZ, as well as the average time spent out of the AZ are of the order of $15 - 20$ minutes, indicating a rather dynamic setting. On one side, this dynamism might make the spreading of the content faster and more efficient. Nonetheless, it may also make it easier for the content to stop floating and disappear from the AZ.
One factor which contributes to the speed of the content diffusion process is the time necessary for a successful transfer of the content from one host to another. In our tests, 10% of transfers required less than 2 seconds, while for 80% of the contents the transfer time has been within 8 seconds (as shown in Fig. 5.4). Given the mobility features of the hosts in the considered setting, these values of transfer time are low enough to have an overall negligeable impact on the performance of the FC service. Indeed, as we have seen, average sojourn times in the AZ are significantly longer than 8 seconds. As we will show, this gives each floating content enough time to reach the vast majority of the hosts in the AZ within its floating time.

An important performance metric in several FC applications is the time taken by a fresh user entering the AZ to get the floating content or, for a user already in the AZ

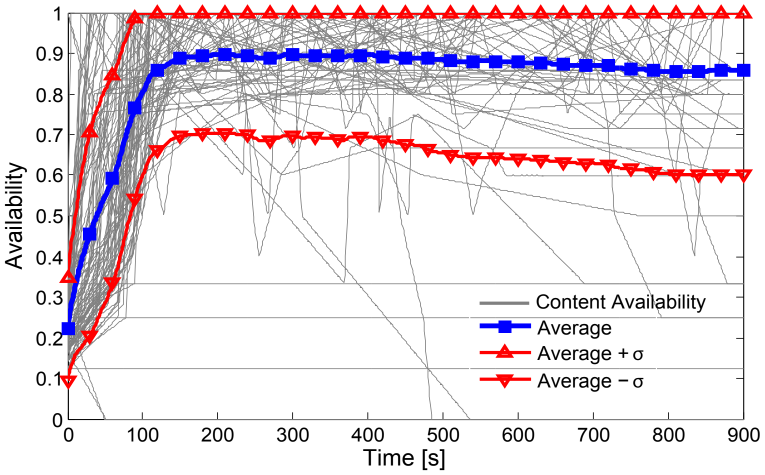


Figure 5.7: Content availability over the content lifetime.

when the content was generated, the time the content takes to reach the user. Especially in dynamic settings, for applications such as emergency warnings, or situated introductions, receiving a new content in a reasonably short time (of the order of few minutes) is essential for an acceptable application performance. In Fig. 5.5 we plot the CDF of the time to get the content. These plots have been scaled to consider as infinite the time for getting those contents which are not delivered to a given user within the content lifetime (that is, 900 seconds). Therefore these plots also indicate how successfully the content has been transferred during its whole floating time. In our experiments, more than 60% of the contents has been received within two minutes, and around ten minutes have been sufficient to get 80% of the contents in the AZ. These values show that in these settings the performance of the FC service has proven adequate for the majority of the FC applications proposed so far.

Fig. 5.6 shows, for each content generated during our tests, the success ratio (the fraction of users inside the AZ who got that content during its floating time), and the number of "replication opportunities" during the content floating period (given by the number of hosts present in the AZ at the time of the content generation, plus all those who entered the AZ during its lifetime).

In order to characterize the empirical distribution of the success ratio, contents have been ordered according to increasing values of this parameter. We can see that about 80% of the contents have a success probability greater than 75%. In order to check to which extent the differences in content success ratio among hosts might be due to mobility patterns and to fluctuations in the population of hosts in the AZ over time, in Fig. 5.6 we also show, for each content, the number of replications opportunities during its floating lifetime. These plots allows to reject the hypothesis of a correlation between the values of success ratio of a given content measured in our experiments, and replication opportunities of that content. A system parameter which has a direct impact on the probability for a user in the AZ to receive a content in a given time interval is the *availability* of that content during the time interval, i.e., the percentage of hosts in the AZ which, in that time interval, have the content. This is confirmed by Fig. 5.7, which shows the evolution of average content availability over the floating lifetime. As expected, the variation of content availability over time is clearly correlated with the time taken to get the content seen in Fig. 5.5, and the time it takes for the average availability to reach its “steady state” value is close to the average time it takes to get a content in the AZ. In Fig. 5.7 we have also plotted per-content availability over time, and the standard deviation, in order to give an idea of the variability of availability across contents, mainly due to the effect of mobility patterns. We can see that only 3.5% of the contents stops fluctuating before 15 minutes, and half of these contents disappear within the first minute after its generation. In other terms, the vast majority of the contents floated successfully for the whole duration of their lifetime.

5.3 Summary

In this chapter, we analyzed the performance of FC service by developing and deploying an Android mobile application in an office environment. To the best of our knowledge, this was the first ever experimental evaluation of FC service in an office setting. We characterized the performance in terms of success ratio, availability and by the time users receive a new content. In an office setting, our results showed that 80% of the users recieve the content within 10 minutes of content generation time. It is a clear indication that contents reach users quite quickly. Moreover, 80% of the contents have a success ratio of greater than 75% showing that a high percentage of contents reach most of the nodes in the network. We also observed that only 3.5% of the contents stopped floating before their lifetime showing that FC is capable of guaranteeing content persistence as well. All these results confirm the suitability of FC as a communication service for context-aware applications.

Chapter 6

Experimenting with Floating Content in a Campus Environment

In this chapter we present an experimental study of the operation of floating content in a university campus context. We describe the development of a smartphone app for this experiment, as well as the experimental setting used for the collection of real data traces. We present and discuss a selection of the results that were collected over one week of operation of FC. We characterize some critical performance aspects which are relevant for applications based on FC, such as content persistence, availability and efficiency with which the content is accessed by users. Finally, we propose a simple analytical model which accurately captures the key behaviors that emerged from the experiment.

6.1 Experimental Setup

In this section we describe our experience with FC services in a large university campus scenario. The experimental FC services were running on *Floaty UC3M*, which was a modified version of the app that we presented in Chapter 5.

6.1.1 Floaty UC3M: a Floating Content Mobile App for Android

In order to perform our tests, we implemented the FC service on an Android smartphone application, named *Floaty UC3M* [50]. We chose to implement the FC app on Android because of its wide diffusion among existing mobile users. Floaty UC3M implements opportunistic communications over Bluetooth, which we selected because it is presently available in almost all smartphones in the market. WiFi Direct was the possible alternative to Bluetooth, but it was discarded because it typically consumes much more energy [49], and poses severe technical problems, due to the lack of transparent user authentication modes.

Contents generated by the app are lightweight. They are composed of an identifier of the device which generated the content (its MAC address and Bluetooth name), a timestamp (the time at which the content was generated), and a sequence number. As a consequence of the small size of the content, during the whole experiment, a single Bluetooth message was sufficient to transfer all of the contents that were transmitted from one node to another. This allowed us to neglect the effect of content size over transfer time, and, in general, over the performance of the FC service during the experiment.

When a device running the app is in the AZ, the app generates a new content every 15 minutes, and contents are transferred to all the other nodes which come within the radio transmission range, and which are running the same app. A content does not expire until the end of the day when it was generated, so every day we generate a fresh population of contents. Note that the availability of several contents generated by each user during its sojourn inside the AZ allowed the test of FC services performance with a fine granularity over time and space in the experimental setting.

In order to detect the presence of floating contents, every 60 seconds the app performs a Bluetooth scan, and builds a list of available Bluetooth peers running the same app.[1] Whenever the list of peers is nonempty, the app chooses one of them, and tries to establish a Bluetooth connection with it. If it succeeds, the two peers exchange all contents they store. However, due to Bluetooth limitations, the app connects to one peer at a time, and spends a few seconds to transfer the available contents between connected peers (up to 12 seconds in our experiments, almost 100% of which due to Bluetooth connection protocols, rather than to data transfer, as we have experimentally verified). In our tests, 18% of transfers required less than 2 seconds, while for 90% of the contents the transfer time has been within 8 seconds (as shown in Fig. 6.1).

Therefore, it is important to optimize the choice of the peers, to improve content diffusion in dynamic settings. To this end, the app ranks peers in radio range according to the time of the last content exchange, starting from the oldest ones, so that the peers that were never met before have absolute precedence.

For the FC mechanism to operate correctly, the most relevant information is whether a user is inside the AZ, rather than its exact position in space. Therefore, instead of exploiting GPS signals or the like, the app just makes use of a coarse localization method based on WiFi signals. Specifically, to decide whether a node is inside the AZ, every 60 seconds the app scans for signals of surrounding access points. If it detects at least

[1] This is possible thanks to the fact that Bluetooth allows the announcement of running services.

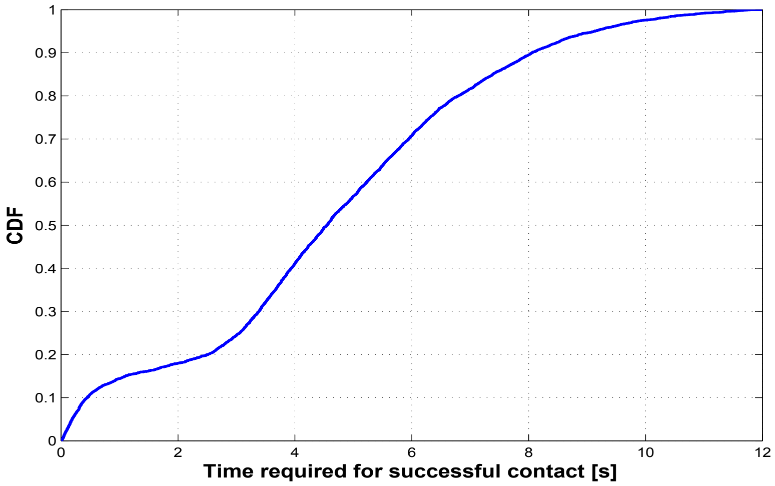


Figure 6.1: CDF of the time required for successssful transfer.

one reference access point (any of the ones present on campus), then it assumes to be inside the AZ. Therefore, our experiment covers a slightly larger area than the university campus. However, being the campus quite large, and being the access points located inside the buildings, the difference is negligible.

All scanning intervals and content generation intervals have been chosen so as to achieve a reasonable trade-off between the performance of FC services and the energy consumed by the devices. In particular, we have quantified the average energy consumption for a test set of smartphones in $\sim 5\%$ battery discharge per hour. Generating contents more frequently, or scanning WiFi and Bluetooth on a sub-minute basis would have made terminals incur excessive battery costs. However, we have experimentally evaluated our test set of devices using different scanning and content generation intervals. As a result, we have found that, given the slow dynamics of the campus environment, the performance of FC services was not benefiting from higher scanning frequencies or more frequent content generation.

The app logs all events related to content transfer between nodes, including timestamps, connection durations, lists of transferred contents. The app also logs all events relating to AZ ingresses and egresses, as well as all failed message transfers. These logs are periodically uploaded to a remote server.

6.1.2 The Experiment

After having tested the Floaty UC3M app with a reduced test set of terminals and users, we advertised the experiment on campus, trying to convince students, researchers, and professors to download and install the app on their smartphones and tablets. In particular, we had access to the Computer Science building, where we were allowed to use fliers and mailing lists, and to drop by classrooms at the end of classes to talk with students. In total, among the people frequenting the building, 62 volunteered to download and install the app on their personal devices. However, only 48 devices produced valid logs, which were used for computing the different statistics shown in the rest of the chapter. Moreover, a tiny fraction of these users never exchanged any content during the whole duration of the experiment. The experiment lasted 5 days (Monday to Friday) from 8 AM to 6 PM, in March 2014. According to the logs that reached our server, during the whole experiment, users generated 1117 contents.

As we will show in Section 6.2, we verified that users had a peculiar mobility pattern, i.e., they moved between home and the university, and, once on campus, they had the tendency to form stable groups and spend long intervals in the same place (as expected for students attending classes). The result of such mobility pattens is scarce spatial interaction with most of the other users. Indeed, this is a trivial but important aspect of the performance of FC services. In this regard, the experiment confirmed that infrastructureless FC is a reasonable choice even for applications meant for users with reduced spatial interactions. However, for our evaluations, we considered only those contents that got a chance to be replicated at least once, which consists in a population of 923 contents.

6.2 Experimental Results

In this section we report the main results of our experiments. We first use the collected logs to characterize the behavior of users participating in the experiments, in terms of mobility, connectivity, and generation of contents. Afterwards, we show the performance figures achieved by the FC service, mainly in terms of content availability and success ratio. For the definitions of both of these performance parameters, please refer to section 5.2.

6.2.1 Characterization of User Behavior

During the experiments, users dynamically enter and exit the AZ, and generate contents only when they are within the AZ, so that the number of nodes present in the AZ, as well as the number of contents floating in the AZ changes over time. Fig. 6.2 illustrates the dynamics of number of nodes and contacts observed in the AZ over the day, averaged over the logs collected during the five days of experiments. For ease of presentation,

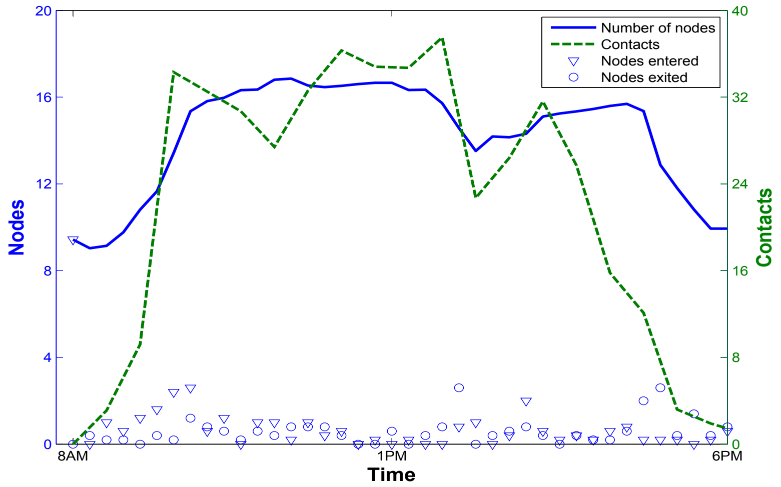


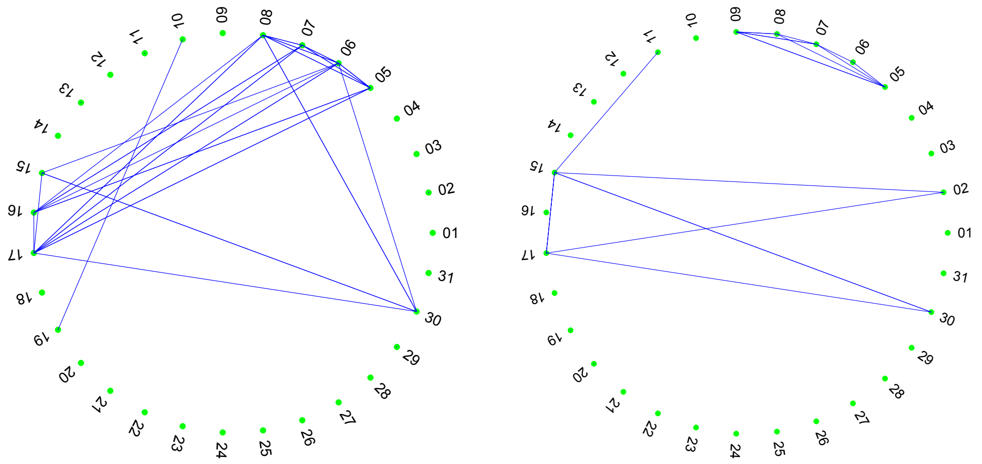
Figure 6.2: Behavior of users over a day (average over 5 days).

arrivals, departures, number of nodes, and number of contacts are counted with a time resolution of 15 minutes.

From the figure, it emerges that the average number of nodes does not change drastically over the day, although it is possible to identify five regions in the graph: (i) the leftmost part of the curves account for the starting of a new day, when arrivals are more numerous than departures (8AM - 10AM); (ii) from 10AM to 2PM, the number of nodes in the AZ changes very little, after which (iii) the curve shows a drop, corresponding to lunch time (2PM - 3PM); (iv) from 3PM to 5PM the number of nodes is again practically stable, while (v) after 5PM there is a prevalence of users leaving the campus and thus the AZ.

The number of contacts between nodes does not follow exactly the same trajectory as the number of nodes, because users within the AZ often move in groups, attend classes, meet in shared areas and cafeterias, so that the connectivity pattern is more irregular over time. In particular, note that nodes present in the AZ at the beginning and at the end of the day do not experience many contacts, which is a symptom of scarce mobility of users arriving early and/or leaving late (probably these are professors or researchers sitting in their offices, where they are under campus WiFi coverage). In general, Fig. 6.2 shows that contacts are, on average, not very frequent. For instance, about 16 users experience in total 33 contacts at 1PM (i.e., between 1PM and 1:15PM), which is a symptom of limited mobility on the time scale of minutes.

Fig. 6.3 gives more insight on the occurrence of contacts between users. In this figure each point corresponds to a node, and a link represents one or more contacts between two nodes during the considered hour of the day. The figure depicts two snapshots of the connectivity graph between users in the AZ. Every link between two users means



(a) 11 AM to 12 PM.

(b) 1 PM to 2 PM.

Figure 6.3: Connectivity graph for different times of the day.

that those users had a contact in the considered time interval, in the first day of the experiment. Clearly, connectivity exhibits drastic changes over the day, which can affect the performance of FC services in terms of users that can be reached, and probability of the content floating within the AZ. Notably, the figure unveils that one or two components are practically sufficient to cover the full set of users experiencing contacts, so that we can infer a *clusterization* of user contacts in which two clusters are practically enough to describe the process. Indeed, one of the most prominent differences between the main mobility models which have been applied to the analysis of FC, and mobility patterns in a realistic experimental setting is the fact that the latter generate nonuniform user distributions in space. As it is common experience, confirmed by the logs of contacts we collected, real user mobility is characterized by correlation in locations and trajectories of users, often resulting in the formation of clusters.

Our logs also show that only 85% of the contacts generate a successful transfer of contents. This behavior is due to users trying to exchange contents while actually moving, e.g., walking through a corridor, and, in general, when there is not enough time to complete the Bluetooth connection procedure and the actual message exchange. Indeed, considering that the rate of unsuccessful content transfers is low, and that the number of contacts logged in our experiments is limited and shows clusterization effects, we infer that, in our experiments, users tend to move quickly from a place to another and then spend long intervals at their destination (as expected for students attending classes on campus).

To further characterize the mobility pattern of users, we plot the CDF of the sojourn times of the users within the AZ in Fig. 6.4. From the figure, it emerges that about 50%

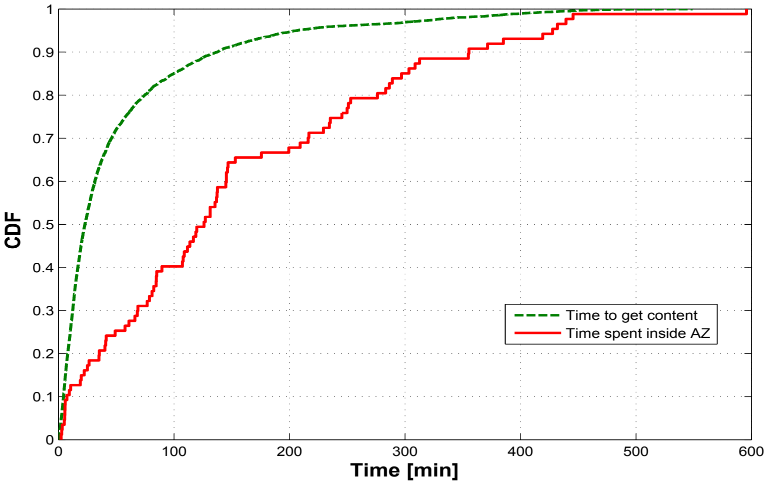


Figure 6.4: Empirical CDFs of user sojourn time and of time to get content.

of the users spend 2 hours or less inside the AZ before leaving (although they possibly come back later, e.g., after lunch). The figure also reports the CDF for the time to get a content once a node enters the AZ, or after a new content is generated while a node was already in the AZ. Notably, most of the contents are replicated within a few tens of minutes (70% of those nodes which get a given content do it within 40 minutes), i. e., much faster than the typical time spent by a node inside the AZ. Therefore, we can conclude that, notwithstanding the limited frequency of contacts, the mobility of users is still suitable to make the contents float within the AZ. We empirically validate this statement by analyzing the floating behavior of contents in the next subsection.

To conclude the empirical evaluation of the user behavior, we report a comparison between the empirical CDF of content lifetimes, and the corresponding ideal CDF for the case of constant number of nodes in the AZ generating contents at regular intervals in Fig. 6.5. Notably, the two CDFs do not differ significantly, although it is possible to notice in the experimental CDF a bias (higher slope in the curve) in favor of the number of contents with lifetime longer than 5 hours and shorter that 8 hours. In general, from our experiments we note that the longer a content floats, the lower the probability that the content will die out early (of course, all contents will die out at the end of the day, so no content can last more than 10 hours in our experiment).

6.2.2 FC Performance Evaluation

In Section 6.2.1 we have already mentioned the fact that the time to get a content is typically short with respect to the time spent by a user in the AZ, and that users exhibit

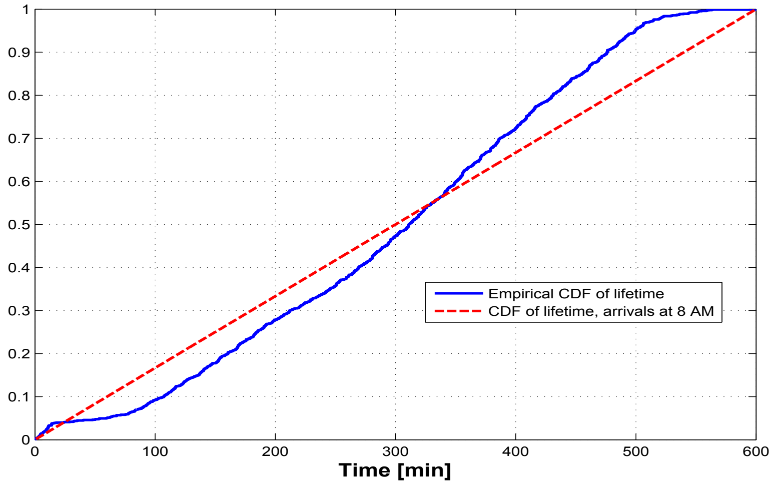


Figure 6.5: CDF of content lifetime compared with the ideal case of uniform arrivals of users and content generation.

scarce mobility on short timescales. However, from the perspective of an application using FC, one of the most studied features of the service, perceived as one of the main measures of its feasibility, is content floating lifetime. Indeed, in a real scenario, all contents naturally disappear at some point in time, not only because of day/night mobility patterns, but also because of stochastic fluctuations in population density and in mobility pattern. Indeed, we noticed that among all contents generated over the five days of the experiment and replicated at least once, only 5% died out before the end of the day. For those contents that do get replicated, but do not reach the end of the day, Fig. 6.6 shows that the vast majority disappears in the early stages of the diffusion process, while the number of floating content replicas has not reached a critical mass. Conversely, when the number of replicas gets *sufficiently high*, the replication mechanism proves efficient enough to compensate content replicas lost because of users leaving the AZ.
This observation allows us to compare our experimental data with results from available models. For mobility patterns such as the ones in our experiment, with relatively few "on-the-fly" content exchanges, from [51] the mean number of nodes in range of a given node should be larger than 1.19 for content to float indefinitely with high probability. However, despite the very low content mortality in our experiments, in our settings the mean number of nodes in range has been 0.612, well below the criticality condition.

The net result of the content dynamics described above is that the average availability of contents grows over time, counting from the content generation time, as shown in Fig. 6.7. In this figure, per-content availability for all contents generated during our

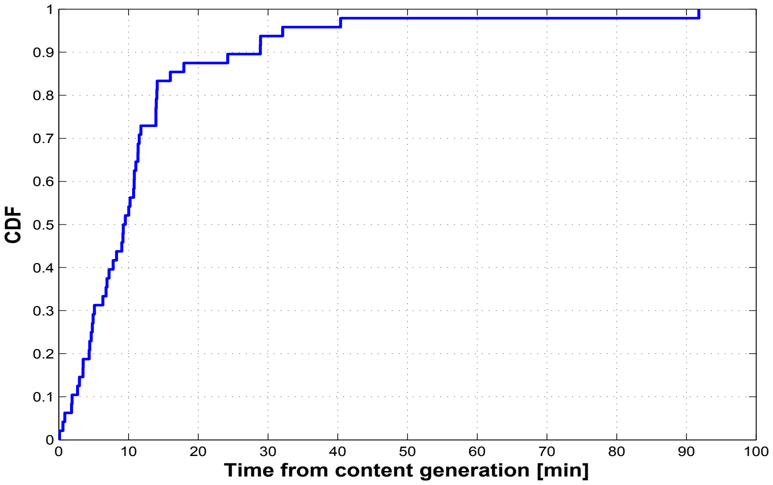


Figure 6.6: Empirical CDFs of content lifetime, conditioned to content being replicated at least once, and dead before 17.00.

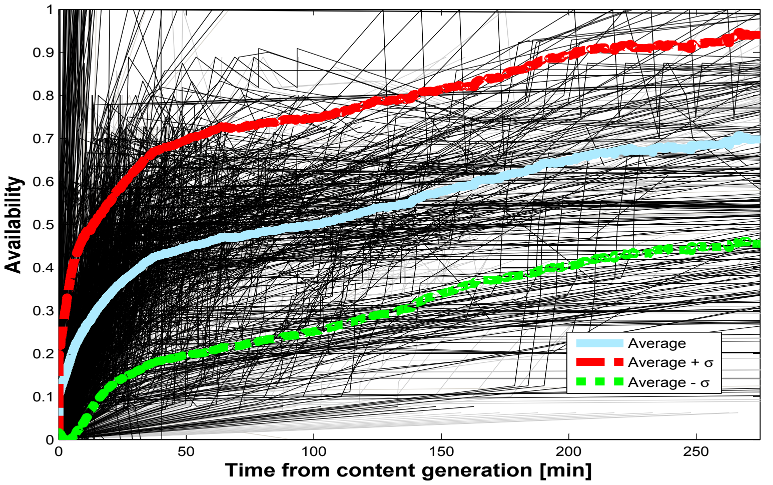


Figure 6.7: Mean content availability over content lifetime, for one seeder.

experiments is plotted in the background. Indeed, one of the main indicators of the effectiveness with which content fluctuates in the AZ is availability, i.e., the fraction of users with a copy of the content. Observing Fig. 6.7, content spreading appears to occur at two speeds over content lifetime. A quick initial diffusion phase is followed by a phase of slower but steady spreading among the rest of the users. This behavior is supported by our clusterization hypothesis, since diffusion within a cluster of contacts would be fast, while spreading a content across clusters would take longer, due to the limited mobility of users.

As a second observation on the empirical availability depicted in Fig. 6.7, we notice

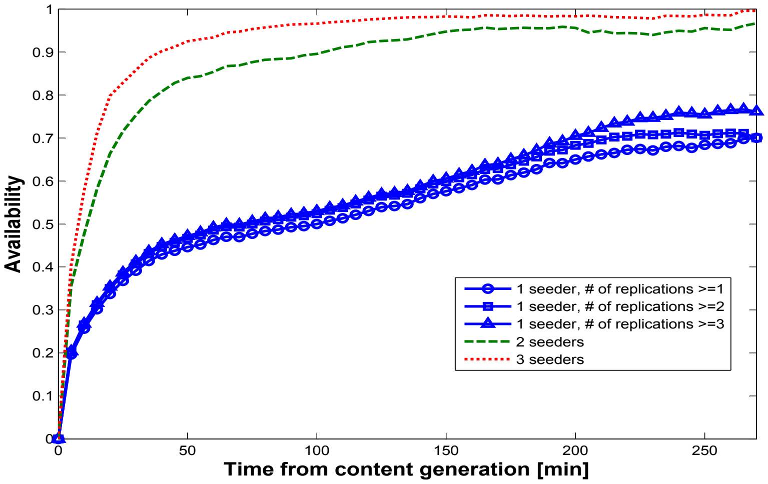


Figure 6.8: Mean content availability over content lifetime, for different number of seeders.

that while contents float, only about half of the users in the AZ possess a copy of the floating contents. To validate the hypothesis that also this behavior is due to the occurrence of clusters, we parse once again our logs to build the following functions of time: the availability of contents replicated at least once, the one of contents replicated at least twice, and the one of contents replicated at least three times. Furthermore, we build the availability of pairs and triples of contents, i.e., the availabilities obtained by considering the joint availability of respectively two or three contents generated roughly at the same time. In this way, we simulate the presence of multiple *seeders* for a same content. The resulting curves are reported in Fig. 6.8. The figure confirms the non-uniformity in user distribution in space. In fact, multi-seeder curves exhibit high availability levels, while little difference can be notice between the singe-seeder curves. In practice, since seeders are chosen at random among all users present in the AZ at the time of content generation, having more seeders improves the probability to reach more clusters within a given time from content generation. In contrast, if the initial replicas reside on users which are in range of each other, the effect on availability is negligible. The little difference between the case with two seeders and the one with three seeders tells us that the mobility pattern in our experimental settings gives rise to a clustered structure that could be modeled with as few as two clusters.

To complete the performance evaluation of FC services in our experiments, we finally report results on the success ratio. This is a measure of the effectiveness in accessing the content stored (floating) in the AZ, as reported in Fig. 6.9 for the cases of one, two or three content seeders. For a given content, the figure shows the ratio between the total number of users who got the content and the total number of users who were present in the AZ (or

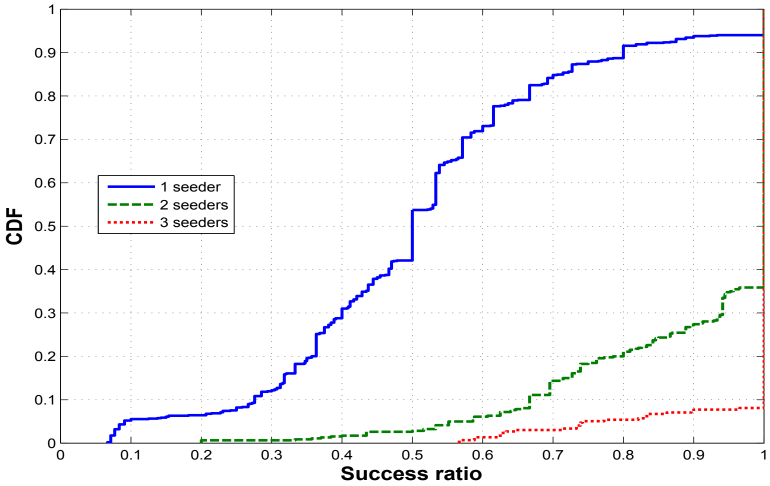


Figure 6.9: Experimental CDF of success ratio, for different number of seeders.

entered the AZ) during the content lifetime. Users who entered the AZ more than once during the content lifetime are counted multiple times. In fact, each time a user exits the AZ, it discards all contents, and if it re-enters the AZ, it tries to collect again all contents floating in the AZ. Specifically, the figure reports the empirical CDF of the success ratio computed over all the 923 contents considered for the experiment statistics. For the case of one single seeder, the peculiar mobility pattern of the users does not allow high success ratios with high probabilities. In fact, half of the contents have a success ratio of 0.5 or less, and just 10% of the contents exhibit a success ratio higher than 80%. This result was somehow expected, since we knew from Fig. 6.7 that the average availability over time does not reach values higher than 70%. With more seeders, both availability and success ratio are boosted. In particular, with two seeders, 65% of the contents are replicated to all users present in the AZ during the content lifetime. With three seeders, all users in the AZ, independently from the time spent in the AZ, receive a copy of more than 90% of the contents.

So far we have empirically computed performance figures for FC. Whether such performance is suitable or not to allow for the diffusion of FC services is out of the scope of our work, and it mostly depends on the requirements of the specific service to be implement on top of FC. However, with our experimental results we have also unveiled that, if required by the application, using just a few independent seeders can dramatically increase both availability and success ratio, so as to reach the needed performance level. For instance, in our campus experiments, no more that three seeders are needed to make FC extremely effective.

Next, we use the lesson learnt from the experiments to build a reliable model for evaluating the performance of FC.

6.3 The Poisson Jumps Model

Here we present a new analytical model for the computation of the main performance parameters of FC services in areas like campuses and large office areas; the analysis is based on a mobility model which reflects typical patterns of such environments, as characterized in the previous sections.

We consider mobile nodes in $\mathbb{R}^2$. The node mobility is described with the *Poisson Jumps* model, derived from the Random Waypoint model by assuming users to move at infinite speed. According to this mobility model, users "jump" from a location to another, staying in each location for a given *stopping time*, which we model as a random variable with mean $1/\mu$. Therefore, a sojourn in the AZ is modeled as a sequence of jumps into a set of locations within the AZ, and a sequence of stops at those locations. Note that this model naturally applies to settings where users move "too fast" for content exchange to take place on-the-fly, and where most of the time is spent on a seat (e.g., for our on campus experiments, in office or in class).

In accordance with experimental data, we assume the mean stopping time to be much longer than the average time for content transfer. Thus, transfers can be considered instantaneous for all practical purposes. We also consider stopping times to be long enough to allow retransmissions in case of failure. Hence, we can neglect the effect of communication errors.

Node arrivals in the AZ follow a Poisson process with intensity γ along time, and they are distributed uniformly in space within the AZ. After each stopping time, a node leaves the AZ with probability $1 - p$. Otherwise, it moves instantly to a new randomly chosen location within the AZ. The resulting spatial distribution of nodes is uniform at any time.

We assume that at time $t = 0$, a "seeder" node generates a content, which has to be spread among nodes within the AZ. The AZ is circular, with radius R, centered at the location of the seeder at $t = 0$. Every node knows its exact position in space at any time. All nodes have the same transmission range r. For $t > 0$, every time two nodes *within the AZ* come in transmission range of each other (we call this a contact), if only one of the two nodes possesses the content, it transfers it to the other. When nodes move outside of the AZ, they delete their own copy of the content. We assume moreover that $R \gg r$, as otherwise direct communications between nodes in the range of the seeder would be enough to spread the content.

As we have seen, one of the main performance parameters for FC is the probability for a user to get the content during its sojourn in the AZ, since this quantity measures the efficiency with which the floating content service makes information available to intended users. The following theorem derives an expression for it.

Theorem 2 (*Success Probability*)**.** *Consider a content floating in an AZ with radius R, in which nodes arrive at a rate γ. Nodes have a transmission range $r \ll R$, and move according to the Poisson Jumps model, with probability $1-p$ of jumping out of the AZ. In the stationary regime, the probability for a node entering the AZ to get the content during its sojourn in the AZ is given by*

$$P = \frac{p_{jump}}{1 - p(1 - p_{jump})}, \tag{6.1}$$

where p_{jump} is the probability of getting the content after a jump:

$$p_{jump} = 1 - e^{-r^2 \frac{\overline{n}}{R^2}}, \tag{6.2}$$

with $\overline{n} = \frac{\gamma}{\mu(1-p)} - \min\left(\frac{\mu(1-p)}{\nu}, \frac{\gamma}{\mu(1-p)}\right)$*, and* $\nu = \frac{2\mu r^2}{R^2}$.

For the proof, please refer to Appendix B. In practical settings, transients (initial and final), as well as fluctuations in the population of users, impact the performance of the FC service, in a measure which depends on content lifetime, and on many other system parameters. Nonetheless, the expression of Theorem 2 is a reasonable estimate of the success ratio of a given content in practical settings, and it allows estimating the impact of various system parameters on performance.

A similar result can be derived for $P(\tau)$, i.e., the probability of getting content after τ seconds spent in the anchor zone. The resulting expression is a function of the CDF of stopping times.

6.4 Model assessment

In order to estimate the user mobility characteristics in the considered settings, so as to parametrize correctly the Poisson Jumps model, we have estimated the "stopping times" of each user, i.e., the amount of time each user spends in a given location. This has been done by exploiting the fact that every time a user terminal produces a new content (every 15 minutes in our on campus experiments), it tries connecting to all user terminals in its range in order to replicate the content. For each terminal, the log of successful

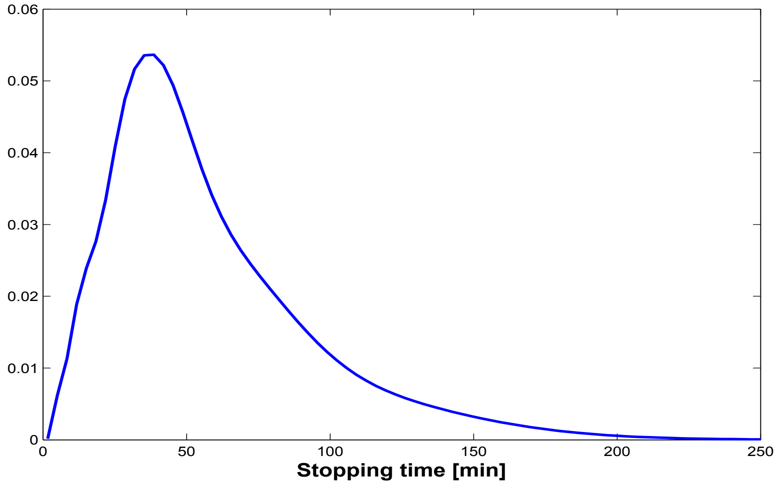


Figure 6.10: Empirical pdf of stopping times estimates.

connection events provides then a fairly accurate picture of the peers around him, at the moment of content generation. The connection latencies induced by Bluetooth act as a lowpass filter for user dynamics, filtering out all peers which remain in range of a given user terminal for less than 12-15 seconds.

The heuristic used to estimate stopping times is based on the assumption that users form small *clusters*, i.e., groups of users all in range of each other. A cluster may correspond or not to a specific location in space (for instance, a classroom, or a group of students chatting in a corridor). All that matters for the purpose of parameterizing the Poisson Jumps model is how long a user is in range of a given set of peers. Of course, in very large groups of people, not all users in the cluster are in range of all the cluster members, but this is a reasonably good approximation for the range of user densities in our experimental setting. More specifically, in our experiment, a cluster in a given time interval has been identified as a set of at least two users, each of which had at least two exchanges of content(s) during the given time interval. As every user periodically produces new content, all members of a cluster frequently exchange content directly among them. Hence, from the logs of those connections it is possible to estimate the composition of the cluster over time.

The arrival/departure of a user in/from a cluster is estimated from the first/last time when that user has established a connection with any peer in that same cluster. As this estimate is built by aggregating connection logs from all users in the cluster, the larger the cluster, the more accurate the estimate, as content generation times are not synchronized across users. This method of estimating stopping times is not able to reliably capture relatively fast dynamics of clusters (peers leaving the cluster only for 2-3 minutes, for

Table 6.1: Availability and success probability: model vs. experiments

	Model	Measured
Average Availability	0.50	0.562
Average Success Ratio (Success Probability)	0.47	0.514

instance), but this event should be infrequent in our setting.

In Fig. 6.10 we can see the shape of the empirical distribution of the stopping time, computed as described above. The resulting mean stopping time is 58 minutes and 14 seconds, which is in accordance with the fact that classes last about one hour and most of the clusters form in classrooms.

We have evaluated numerically the success ratio, by assuming a mean sojourn time of about $1h$ 40 min, computed from our experiments by excluding nodes which entered or exited the AZ towards the end of the day, i.e., after 5 pm. This was done in order to have a more homogeneous set of data from which to extract this parameter. The node arrival rate, averaged over the whole week of the experiment, is 0.0015 nodes per second. The area of the AZ in the experiment has been estimated as about 7700 square meters, for an equivalent radius of about 50 meters, which accounts only for buildings in which classes were given during the experiment. As we can see from Table 6.1, despite its simplicity, our model is able to predict with very good accuracy the performance of the FC service. Its being conservative can be attributed mainly to the assumption of uniform user distribution in the AZ.

6.5 Summary

In this chapter we presented an experimental study of the operation of floating content in a university campus context. We characterized the performance of the service in terms of content persistence, availability and efficiency with which the content is accessed by users. Our analysis highlighted some critical performance aspects which are relevant for applications based on FC. We showed that, even in such a highly dynamic setting, a relatively low user density is enough to guarantee content persistence over time, contrary to predictions from available models. More specifically, we showed that using just one seeder, the special type of user movement in the environment that we considered in our experiment makes the availability of generated contents grow fast in the first hour, reaching on average about 40% of the users in the anchor zone. The successive increase in availability is slower, so that about 60% of users in the AZ is reached on average after four hours. Whether this is acceptable for a floating content service, it depends on the specific application, but we observed that it is easy to reach much higher availability values by just increasing the number of seeders to two. Indeed, by doing so, the number of users reached after 1 hour became about 90% on average, and after four hours was around 98% on average. This provided quite interesting indications for the viability and the implementation of floating content services in university settings, and in general in office environments. Our results suggest that future efforts in FC performance modeling should be based on more realistic mobility models, including the effects of clustering as well as of correlation among users' mobility patterns, and focus on performance over short timescales.

Chapter 7

Conclusions and Future Work

In this thesis our objective was to characterize the performance of context-aware applications that use FC as a communication service. We defined *success probability*, which captures the likelihood for a user to receive the content when traversing the AZ as the primary performance indicator. As a first step, we developed a simple analytical model for understanding the impact of FC design parameters like AZ radius, node transmission range, and node density on the performance of an application that use FC as the communication service. Our simple analytical model helped us to the understand the correlation between the system design parameters of FC and the performance of context-aware applications that use FC as communication service. From a system design perspective, parameters like AZ radius, node transmission range, and node density have a significant impact on the performance of the applications and our analysis can be used to tune key system parameters so as to achieve the desired application performance. A key characteristic of our modeling approach was that success probability was computed from few primitive system parameters allowing our analysis to be generalized to various settings, including different AZ shapes, different user mobility patterns, different user speed distributions, different service and application models. We applied our analysis to estimate the success probability for three representative categories of context-aware applications, and showed how the system can be configured to achieve the application's target. In order to complement our analytical study, we validated our simple model using extensive simulations under different settings. Our simulation results show that our model-based predictions are indeed highly accurate under a wide range of conditions and high success probabilities are achieved with reasonably sized anchor zones even at low node densities, demonstrating the viability of floating content as an enabler for context-aware applications.

As a next step, we investigated the impact of some more realistic end-user mobility models on the performance of context-aware applications that can use FC as a communication service. By considering three different categories of context-aware applications under four different user mobility models, we found that FC is capable of providing very effective performance to a variety of context-aware applications under quite diverse mobility patterns. Comparing simulation results to the performance predictions of a simple analytical model, we found that our model can provide useful performance predictions even for complex and realistic mobility models. For all the considered mobility models, high success probabilities can be achieved by tuning the anchor zone radii for a variety of context-aware applications, which justifies the viability of FC as an enabler for context-aware applications.

Then we extended our previous studies of the performance of FC by doing real world experiments. It was very important to evaluate the performance in a real world setting because the previous works were based on simplifying assumptions for user mobility, and on a simplified model for data exchange. As a result, the resulting models gave indications about the potential and limitations of FC, but they were only first-order predictions of performance in realistic settings. Indeed, important aspects, such as the combined impact of the actual mobility patterns of users, of the communication protocols, of the specific propagation characteristics in the chosen area, of localization accuracy, were difficult to investigate analytically and to evaluate by simulation. Moreover, it was important to understand the impact of the limitations and of the specific features of available protocols for opportunistic communications (such as Bluetooth and WiFi Direct) on the performance of the FC service. For conducting real world experiments, we developed and deployed an Android mobile application based on FC in an office environment. To the best of our knowledge, this was the first ever experimental evaluation of FC service in an office setting. Our results confirmed the suitability of FC as a communication service for context-aware applications in an office setting.

Finally, we extended our experimental evaluation of the FC communication paradigm by deploying our Android based mobile app in a campus/large office setting. From this experiment, we characterized the mobility patterns, and we assessed the performance of services implemented using the FC paradigm. Our results unveiled the key relevance of group dynamics in user movements for the FC performance. Surprisingly, in such an environment, our results showed that a relatively low user density is enough to guarantee content persistence over time, contrarily to predictions from available models. We also investigated the impact of multiple seeders on the performance and we observed that it is easy to reach much higher values for success probability by just increasing the number of seeders to two. Based on these experimental findings, we developed a novel simple analytical model that accounts for the peculiarities of the mobility patterns in such a setting, and that can accurately predict the effectiveness of FC for the implementation of services in a campus/large office setting.

This work can be extended into two different directions. The first direction involves extending the analytical model to various other mobility models. Our analytical model is quite flexible and other mobility models can be easily incorporated without changing the structure of the formulas. Another possible extension of the model involves incorporating a more realistic communication model into the analytical model. From the experimental implementation standpoint, new communication technologies like Wi-Fi Direct can be explored. Wi-Fi Direct provides quite interesting opportunities for opportunistic communications due to its longer range and higher data transfer rate as compared to Bluetooth. At the time of writing of this thesis, implementing FC using commercial off-the-shelf devices supporting Wi-Fi Direct was not possible due to the lack of user transparent authentication modes. Moreover, it was also not supported by all the Android devices currently available. However, it is expected that in near future, Wi-Fi Direct will be supported by all the new devices and its API will also allow the developer to exploit transparent authentication modes. Therefore using Wi-Fi Direct, new and interesting context-aware applications based on FC can be developed.

References

[1] A. Zimmermann, A. Lorenz, and R. Oppermann, "An operational definition of context," in *CONTEXT'07*, 2007, pp. 558–571.

[2] A. K. Dey, "Understanding and using context," *Personal and Ubiquitous Computing*, vol. 5, pp. 4–7, 2001.

[3] S. Dhar and U. Varshney, "Challenges and business models for mobile location-based services and advertising," *Commun. ACM*, vol. 54, no. 5, pp. 121–128, May 2011.

[4] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, ser. MCC '12. New York, NY, USA: ACM, 2012, pp. 13–16.

[5] B. Nunes, M. Mendonca, X. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: Past, present, and future of programmable networks," *Communications Surveys Tutorials, IEEE*, vol. PP, no. 99, pp. 1–18, 2014.

[6] D. Evans, "The internet of everything - how more relevant and valuable connections will change the world," in *CISCO IBGS (White Paper)*, 2012.

[7] B. Liu, B. Khorashadi, D. Ghosal, C.-N. Chuah, and M. H. Zhang, "Assessing the vanet's local information storage capability under different traffic mobility," in *INFOCOM'10*, 2010, pp. 116–120.

[8] A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, "The anatomy of a context-aware application," *Wirel. Netw.*, vol. 8, no. 2/3, pp. 187–197, Mar. 2002. [Online]. Available: http://dx.doi.org/10.1023/A:1013767926256

[9] T. Gu, H. K. Pung, and D. Q. Zhang, "A service oriented middleware for building context-aware services," *Journal of Network and Computer*

Applications, vol. 28, no. 1, pp. 1 – 18, 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804504000451

[10] M. Raento, A. Oulasvirta, R. Petit, and H. Toivonen, "Contextphone: a prototyping platform for context-aware mobile applications," *Pervasive Computing, IEEE*, vol. 4, no. 2, pp. 51–59, Jan 2005.

[11] P. Fahy and S. Clarke, "Cass: a middleware for mobile context-aware applications," in *Workshop on Context Awareness, MobiSys*, 2004.

[12] D. Siewiorek, A. Smailagic, J. Furukawa, A. Krause, N. Moraveji, K. Reiger, J. Shaffer, and F. L. Wong, "Sensay: A context-aware mobile phone," in *Proceedings of the 7th IEEE International Symposium on Wearable Computers*, ser. ISWC '03. Washington, DC, USA: IEEE Computer Society, 2003, pp. 248–. [Online]. Available: http://dl.acm.org/citation.cfm?id=946249.946884

[13] L. Capra, W. Emmerich, and C. Mascolo, "Carisma: context-aware reflective middleware system for mobile applications," *Software Engineering, IEEE Transactions on*, vol. 29, no. 10, pp. 929–945, Oct 2003.

[14] H. Ogata and Y. Yano, "Context-aware support for computer-supported ubiquitous learning," in *Wireless and Mobile Technologies in Education, 2004. Proceedings. The 2nd IEEE International Workshop on*, 2004, pp. 27–34.

[15] K. Henricksen and J. Indulska, "Developing context-aware pervasive computing applications: Models and approach," *Pervasive and Mobile Computing*, vol. 2, no. 1, pp. 37 – 64, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1574119205000441

[16] J. E. Bardram, "Applications of context-aware computing in hospital work: Examples and design principles," in *Proceedings of the 2004 ACM Symposium on Applied Computing*, ser. SAC '04. New York, NY, USA: ACM, 2004, pp. 1574–1579. [Online]. Available: http://doi.acm.org/10.1145/967900.968215

[17] V. Cozzolino, "Design and implementation of an android context-aware application based on floating content," Master's thesis, Universita' Federico II, Napoli, 2013. [Online]. Available: http://eprints.networks.imdea.org/id/eprint/607

[18] A. Sayed, A. Tarighat, and N. Khajehnouri, "Network-based wireless location: challenges faced in developing techniques for accurate wireless location information," *Signal Processing Magazine, IEEE*, vol. 22, no. 4, pp. 24 – 40, July 2005.

[19] S. Ali, G. Rizzo, V. Mancuso, V. Cozzolino, and M. Ajmone Marsan, "Experimenting with floating content in an office setting," *Communications Magazine, IEEE*, vol. 52, no. 6, pp. 49–54, June 2014.

[20] A. Balasubramanian, R. Mahajan, and A. Venkataramani, "Augmenting mobile 3g using wifi," in *MobiSys '10*, 2010, pp. 209–222.

[21] E. Hyytiä, J. Virtamo, P. Lassila, J. Kangasharju, and J. Ott, "When does content float? characterizing availability of anchored information in opportunistic content sharing," in *INFOCOM*, Shanghai, China, Apr. 2011, pp. 3123–3131.

[22] J. Ott, E. Hyytia, P. Lassila, T. Vaegs, and J. Kangasharju, "Floating content: Information sharing in urban areas," in *PerCom 2011*, March 2011, pp. 136 –146.

[23] J. Virtamo, E. HyytiÃd', and P. Lassila, "Criticality condition for information floating with random walk of nodes," *Performance Evaluation*, vol. 70, no. 2, pp. 114 – 123, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0166531612001204

[24] S. Ali, G. Rizzo, B. Rengarajan, and M. A. Marsan, "A simple approximate analysis of floating content for context-aware applications," in *Proceedings of the fourteenth ACM international symposium on Mobile ad hoc networking and computing*, ser. MobiHoc '13. New York, NY, USA: ACM, Available at: http://eprints.networks.imdea.org/535/, 2013, pp. 271–276. [Online]. Available: http://doi.acm.org/10.1145/2491288.2491321

[25] I. Leontiadis and C. Mascolo, "Opportunistic spatio-temporal dissemination system for vehicular networks," in *Proceedings of the 1st International MobiSys Workshop on Mobile Opportunistic Networking*, ser. MobiOpp '07. New York, NY, USA: ACM, 2007, pp. 39–46. [Online]. Available: http://doi.acm.org/10.1145/1247694.1247702

[26] W. Gao, Q. Li, B. Zhao, and G. Cao, "Multicasting in delay tolerant networks: A social network perspective," in *Proceedings of the Tenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '09. New York, NY, USA: ACM, 2009, pp. 299–308. [Online]. Available: http://doi.acm.org/10.1145/1530748.1530790

[27] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: Social-based forwarding in delay tolerant networks," in *Proceedings of the 9th ACM International*

Symposium on Mobile Ad Hoc Networking and Computing, ser. MobiHoc '08. New York, NY, USA: ACM, 2008, pp. 241–250. [Online]. Available: http://doi.acm.org/10.1145/1374618.1374652

[28] A. Wegener, H. Hellbruck, S. Fischer, C. Schmidt, and S. Fekete, "Autocast: An adaptive data dissemination protocol for traffic information systems," in *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, Sept 2007, pp. 1947–1951.

[29] G. Karlsson, V. Lenders, and M. May, "Delay-tolerant broadcasting," *Broadcasting, IEEE Transactions on*, vol. 53, no. 1, pp. 369–381, March 2007.

[30] O. R. Helgason, E. A. Yavuz, S. T. Kouyoumdjieva, L. Pajevic, and G. Karlsson, "A mobile peer-to-peer system for opportunistic content-centric networking," in *Proceedings of the Second ACM SIGCOMM Workshop on Networking, Systems, and Applications on Mobile Handhelds*, ser. MobiHeld '10. New York, NY, USA: ACM, 2010, pp. 21–26. [Online]. Available: http://doi.acm.org/10.1145/1851322.1851330

[31] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Pocket switched networks: Real-world mobility and its consequences for opportunistic forwarding," University of Cambridge Computer Laboratory, Tech. Rep. UCAM-CL-TR-617, August 2005.

[32] A. Lindgren, C. Diot, and J. Scott, "Impact of communication infrastructure on forwarding in pocket switched networks," in *Proceedings of the 2006 SIGCOMM Workshop on Challenged Networks*, ser. CHANTS '06. New York, NY, USA: ACM, 2006, pp. 261–268.

[33] A. A. V. Castro, G. Di Marzo Serugendo, and D. Konstantas, "Hovering information - self-organising information that finds its own storage," in *SUTC '08*, 2008, pp. 193–200.

[34] N. Thompson, R. Crepaldi, and R. Kravets, "Locus: a location-based data overlay for disruption-tolerant networks," in *CHANTS '10*, 2010, pp. 47–54.

[35] V. Lenders, M. May, G. Karlsson, and C. Wacha, "Wireless ad hoc podcasting," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 12, no. 1, pp. 65–67, Jan. 2008.

[36] I. Leontiadis and C. Mascolo, "Opportunistic spatio-temporal dissemination system for vehicular networks," in *MobiOpp '07*, 2007, pp. 39–46.

[37] O. R. Helgason, E. A. Yavuz, S. T. Kouyoumdjieva, L. Pajevic, and G. Karlsson, "A mobile peer-to-peer system for opportunistic content-centric networking," in *Mobi-Held '10*, 2010, pp. 21–26.

[38] J. Whitbeck, Y. Lopez, J. Leguay, V. Conan, and M. D. de Amorim, "Push-and-track: Saving infrastructure bandwidth through opportunistic forwarding," *Pervasive and Mobile Computing*, 2012.

[39] C. Bettstetter, "Mobility modeling in wireless networks: Categorization, smooth movement, and border effects," *ACM Mobile Computing and Communications Review*, p. 2001.

[40] P. Nain, D. Towsley, B. Liu, and Z. Liu, "Properties of random direction models," in *INFOCOM '05*, vol. 3, March 2005, pp. 1897 – 1907.

[41] Y. Wu, S. Deng, and H. Huang, "Evaluating the impact of selfish behaviors on epidemic forwarding in mobile social networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2013, no. 02, p. P02018, 2013. [Online]. Available: http://stacks.iop.org/1742-5468/2013/i=02/a=P02018

[42] A. Varga, "The omnet++ discrete event simulation system," *ESM 2001*, June 2001.

[43] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, "A group mobility model for ad hoc wireless networks," in *Proceedings of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, ser. MSWiM'99. New York, NY, USA: ACM, 1999, pp. 53–60. [Online]. Available: http://doi.acm.org/10.1145/313237.313248

[44] F. Bai, N. Sadagopan, and A. Helmy, "IMPORTANT: a framework to systematically analyze the Impact of Mobility on Performance of Routing Protocols for Adhoc Networks," in *Proceedings of IEEE INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications*, 2003.

[45] S. Uppoor, O. Trullols-Cruces, M. Fiore, and J. M. Barcelo-Ordinas, "Generation and analysis of a large-scale urban vehicular mobility dataset," *IEEE Transactions on Mobile Computing*, vol. 99, no. PrePrints, p. 1, 2013.

[46] *Cologne mobility traces:*, http://kolntrace.project.citi-lab.fr/.

[47] S. Ali, G. Rizzo, M. Marsan, and V. Mancuso, "Impact of mobility on the performance of context-aware applications using floating content,"

in *Context-Aware Systems and Applications*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, P. C. Vinh, V. Alagar, E. Vassev, and A. Khare, Eds. Springer International Publishing, 2014, vol. 128, pp. 198–208. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-05939-6_20

[48] *Floaty app*, https://play.google.com/store/apps/details?id=com.vittorio.

[49] P. Serrano, A. De La Oliva, P. Patras, V. Mancuso, and A. Banchs, "Greening wireless communications: Status and future directions," *Comput. Commun.*, vol. 35, no. 14, pp. 1651–1661, Aug. 2012.

[50] *Floaty UC3M*, https://play.google.com/store/apps/details?id=com.uc3m.

[51] E. Hyytia, P. Lassila, J. Ott, and J. Kangasharju, "Floating information with stationary nodes," in *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2012 10th International Symposium on*, May 2012, pp. 361–366.

[52] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Performance analysis of mobility-assisted routing," in *MobiHoc '06*, 2006, pp. 49–60.

[53] M. D. Springer, *The Algebra of Random Variables*. John Wiley & Sons, 1979.

Appendix A

Proof of Result 1: In order to prove Result 1, we first introduce the following lemmas.

Lemma 1. *Under the RD mobility model with node density λ, when nodes have a transmission radius of r, and velocity equal to v, the number of contacts made by a node in a time interval τ is Poisson distributed with mean $\mu_C = 2rv\tau\lambda$.*

Proof. Without loss of generality, we consider the perspective of a node i with velocity vector $\vec{v_i} = (v, \angle 0)$. We calculate the relative velocities of all nodes in the system with node i as the reference. Thus, we can consider node i to be non-moving, with the other nodes mobility characterized by their relative velocity vector. The total number of new contacts made by i in a time interval τ is equivalent to the number of nodes that enter a circle of radius r centered at the non-moving node i in that time interval, under the relative velocity model. Under the RD mobility model, nodes never change direction and thus a single node cannot enter this circle multiple times. Nodes choose their directions independently and there are no spatial correlations under the RD model, thus the number of nodes entering the circle in two disjoint time intervals are also independent. Further, the stationarity of the RD model implies that the distribution of the number of nodes entering the circle only depends on the interval τ. Therefore, the number of contacts made by a node in a time interval is Poisson distributed.

In order to find the mean of the Poisson distribution, we integrate the average rate at which nodes cross the boundary of the circle delineating the transmission range. The relative velocity of node j with velocity vector $\vec{v_j} = (v, \angle\theta_j)$ is

$$\vec{v'}_j = \left(2v\sin\left(\frac{\theta}{2}\right), \angle\frac{\pi+\theta}{2}\right)$$

We consider infinitesimal sections of the circumference of the circle, which lie at an angle ψ with respect to the x axis. At each section, we consider the possible directions from which nodes could enter (note that relative velocities vectors have angles between $\pi/2$ and $3\pi/2$), and use the relative velocity to capture the flow from each direction entering the transmission range through that section, The mean number of contacts in the interval τ is:

$$\int_0^\tau \int_0^\pi \int_{\psi+\frac{\pi}{2}}^{\frac{3\pi}{2}} 2v \sin\left(\frac{2\phi-\pi}{2}\right)\cos(\phi-\psi)\frac{\lambda r}{\pi} d\phi d\psi dt$$

$$+\int_0^\tau \int_\pi^{2\pi} \int_{\frac{\pi}{2}}^{\psi-\frac{\pi}{2}} 2v \sin\left(\frac{2\phi-\pi}{2}\right)\cos(\phi-\psi)\frac{\lambda r}{\pi} d\phi d\psi dt$$

$$= 2rv\tau\lambda$$

□

Lemma 2. *In equilibrium state, for a node traversing an AZ under the Random Direction mobility model, the probability of meeting k nodes is given by*

$$\int_0^{2R} \frac{\ell^2}{\pi R^2\sqrt{4R^2-\ell^2}} \frac{(2r\ell\lambda)^k e^{-2r\ell\lambda}}{k!} d\ell \qquad \text{(A.1)}$$

Proof. Let A and B be the entry and exit points of a node traversing an AZ, respectively, as shown in Fig. A.1. Since nodes move in a straight line, the length of the chord AB is given by

$$L(y) = 2\sqrt{R^2-y^2} \qquad \text{(A.2)}$$

where y is the distance |CY|. Due to the properties of the RD mobility model, node distribution is uniform in space at any point in time [40], and therefore y is uniformly distributed in $[0, R]$. Then the pdf of L is given by

$$f_L(y) = \frac{2\sqrt{R^2-y^2}}{\int_0^R 2\sqrt{R^2-y^2}dy} = \frac{4\sqrt{R^2-y^2}}{R^2\pi} \qquad \text{(A.3)}$$

Letting $\ell = L(y)$, and substituting we finally get

$$f_L(\ell) = \frac{f_L(g_1^{-1}(\ell))}{|g_1'(g_1^{-1}(\ell))|} = \frac{\ell^2}{R^2\pi\sqrt{4R^2-\ell^2}} \qquad \text{(A.4)}$$

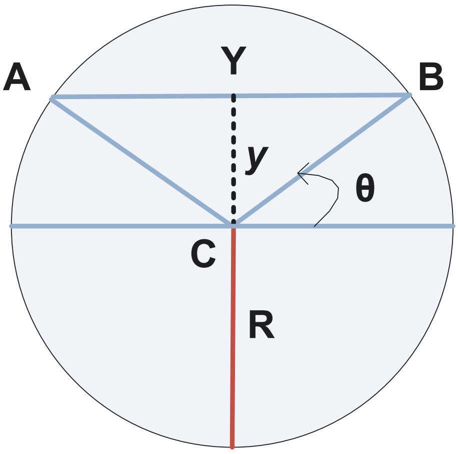


Figure A.1: Chord length in anchor zone

with $\ell \in [0,\ 2R]$.

Using Lemma 1 with $l = v\tau$, the probability of meeting k nodes along this trajectory with a Poisson distribution, with intensity $2r\ell\lambda$:

$$P(meet\ k\ nodes|\ell) = \frac{(2r\ell\lambda)^k e^{-2r\ell\lambda}}{k!} \tag{A.5}$$

So that the probability of meeting k nodes is given by (A.1). □

Lemma 3. *Consider an AZ in equilibrium state, with an average number of nodes equal to $\overline{N}$, and let $\overline{n}$ and $\overline{m}$ denote the average numbers of nodes with and without content $\mathcal{I}$, respectively. Then $\overline{m} = \min(\frac{v}{Q\nu R}, \lambda\pi R^2)$ and $\overline{n} = \lambda\pi R^2 - \overline{m}$, with ν given by $\frac{2rv^2}{(\pi R^2)}$.*

Proof. We denote with $n(t)$ and $m(t)$ the number of nodes with and without content $\mathcal{I}$, respectively, at a given time t, and define $N(t) = n(t) + m(t)$. All these quantities vary over time, as nodes move in and out of the anchor zone, and as content is exchanged. We now build a set of differential equation which describe how these quantities vary over time. Consider the time interval $[t, t+\Delta t]$, and let $\delta n(t) = n(t+\Delta t) - n(t)$, and similarly for $\delta m(t)$. $n(t)$ varies over time due to nodes with content $\mathcal{I}$ exiting the AZ, and to nodes without content in the AZ getting the content.

For the first contribution, we assume that users with content are uniformly distributed within the AZ. Then, the average number of users with content in an area A within the AZ is given by $A\frac{n(t)}{\pi R^2}$. Consider now the time interval Δt: the users with the content that will go out of the AZ are present in the ring of depth $v\Delta t$ around the border of the AZ, whose area can be approximated as $2\pi R v \Delta t$. On average, half of them have a component of their speed in a direction opposite to the center of the AZ (they are moving out), and half in a direction toward the center (they are moving in). We introduce the approximation that all those nodes in the ring who have a component opposite to the center will leave the AZ in the time interval $(t,\ t+\Delta t)$. Putting all together, $\frac{n(t)v\Delta t}{R}$ is the average number of users with content which go out of the AZ by time $t+\Delta t$.

For the second contribution, the frequency ν at which two nodes come within the transmission range r of each other inside an area A, for the Random Direction mobility model, is given by $\frac{2rv^2}{A}$ [52]. The probability that a node picked at random within the AZ has the content $\mathcal{I}$ at time t is given by $p(t) = \frac{n(t)}{N(t)}$, and the probability that content $\mathcal{I}$ is transferred during an event of two nodes coming into contact within the AZ is $2p(t)(1 - p(t))Q$. With $N(t)$ nodes in the AZ, there are $\frac{N(t)(N(t)-1)}{2}$ pairs of nodes that could come into contact. Thus, the average amount of nodes that receive the content $\mathcal{I}$ in the AZ in the considered time interval is given by $\nu\frac{(N(t)-1)(N(t)-n(t))n(t)}{N(t)}Q\Delta t \cong \nu n(t)(N(t) - n(t))Q\Delta t$. Finally, dividing by Δt and letting this time interval go to zero, we get

$$\frac{dn(t)}{dt} = \nu n(t)(N(t) - n(t))Q - \frac{n(t)v}{R} \tag{A.6}$$

The number of nodes inside the AZ without content $\mathcal{I}$ varies in time due to:

- nodes without content $\mathcal{I}$ exiting the AZ,
- nodes entering the AZ (we assume all entering nodes do not have the content, because all nodes delete their copy of $\mathcal{I}$ as soon as they exit the AZ),
- nodes without content $\mathcal{I}$ in the AZ getting the content, by meeting other node(s) with content $\mathcal{I}$.

The first and second contributions can be derived with a similar procedure as used in (A.6) for computing $n(t)$. Therefore, the average number of nodes without content $\mathcal{I}$ exiting the AZ in the considered time interval is given by $\frac{m(t)v\Delta t}{R}$, and the new nodes entering the AZ in the same time interval is given by $\lambda v\pi R\Delta t$. The third term is the same as in the previous equation, with a change in sign. Putting all together, we get the following differential equation for $m(t)$:

$$\frac{dm(t)}{dt} = \lambda v\pi R - \nu n(t)(N(t) - n(t))Q - \frac{m(t)v}{R} \tag{A.7}$$

Since we assume that the system is in equilibrium, the time averages of $m(t)$ and $n(t)$, indicated respectively as $\overline{n}$ and $\overline{m}$, remain constant over time. We can write for $\overline{n}$ and $\overline{m}$ differential equations very similar to those derived above, and we can set both $\frac{d\overline{m}}{dt}$ and $\frac{d\overline{n}}{dt}$ equal to zero. Solving for $\overline{n}$ and $\overline{m}$ we get the expressions in the lemma. □

Lemma 4. *With the same assumptions as before, the probability that a node gets content $\mathcal{I}$ given that it meets k nodes is given by*

$$1-\left(1-\frac{Q\overline{n}}{(\overline{m}+\overline{n})}\right)^{k} \tag{A.8}$$

Proof. The probability for a node to successfully get the content upon meeting another node, can be computed by the product of the probability that the encountered node has the content, (equal to the average fraction of nodes having content in the AZ, and given by $\frac{\overline{n}}{\overline{n}+\overline{m}}$), and the probability of successful information transfer Q. The probability in the lemma is then derived as the probability that at least one out of k encounters with other nodes results into a successful transfer of content. □

Proof. (Result 1) Consider a node traversing the AZ. This node gets content $\mathcal{I}$ if, during its traversal:

- at least one of the encountered nodes has the content, and
- the information is transferred successfully between the two nodes. This implies that the two nodes have been in range for a sufficiently long time to allow the message to be transferred, possibly in presence of transmission errors (collisions and other impairments).

In steady state, the success probability P_s can be written as

$$P_s = \sum_{k=1}^{\infty} P(meet\ k\ nodes)P(get\ \mathcal{I}|\ meet\ k\ nodes) \tag{A.9}$$

The probability of meeting k nodes is given by Lemma 2, while the probability to successfully get the content upon meeting k nodes is given by Lemma 4. Substituting, we get (3.1). □

Proof of Theorem 1:

Proof. Let us consider two nodes A and B with transmission range r, which come in range of each other. The amount of time these nodes will stay in contact is given by L_r/v', where L_r and $\vec{v'}$ are respectively the length of the chord that one node travels within the transmission range of the other, and their relative speed, with $0 \leq v' \leq 2v$. Using a similar approach as for the derivation of (A.4), the probability density function of L_r is given by $f_{L_r}(\ell) = \frac{\ell^2}{r^2\pi\sqrt{4r^2-\ell^2}}$. In the RD mobility model, the probability density function of v' is given by $f_{v'}(\omega) = \frac{2}{\pi\sqrt{4v^2-\omega^2}}$. Using (A.3) and the formula for the pdf of the ratio of two random variables [53], the pdf of the contact duration can be written as (3.3). If X' is the amount of time required for each transfer attempt, then a necessary condition for the information transfer to be successful is to have a contact duration greater than or equal to X'. Including in S all factors relative to communication problems like contention or collisions, we get Eq.3.2 when nodes continually retry upon transmit failures. □

Proof of Result 2: In order to prove Result 2, we first need to introduce the following lemma.

Lemma 5. *In steady state, for a node traversing an AZ and a ROI, under the Random Direction mobility model, the probability of meeting k nodes before leaving the ROI can be approximated as*

$$P_{BL}(met\ k) = \int_{\sqrt{R_2^2-R_1^2}}^{R_1+R_2} \left(f_{L_{AC}}(\ell) \frac{(2r\ell\lambda)^k e^{-2r\ell\lambda}}{k!} \right) d\ell \tag{A.10}$$

Proof. First of all, we compute the average length of the path AC that a node travels inside an AZ and a ROI, before leaving the ROI (as shown in Fig. A.2). Let A be the entry point into the AZ, and C be the point where the node exits the ROI, respectively, then, since nodes travel along a straight line, the length of the chord AC is given by

$$L = |AC| = g_2(y) = \sqrt{R_1^2 - y^2} + \sqrt{R_2^2 - y^2} \tag{A.11}$$

where y is the distance |XY|. Assuming that y is uniformly distributed between 0 and R_1, the pdf for the chord length AC can be computed as

$$f_{L_{AC}}(\ell) = \frac{f_L(g_2^{-1}(\ell))}{|g_2'(g_2^{-1}(\ell))|} \tag{A.12}$$

with $\ell \in [\sqrt{R_2^2 - R_1^2},\ R_1 + R_2]$.

Using a similar approach as in (A.1), the probability of meeting k nodes is given by (A.10).

□

Proof. (Result 2) To compute the success probability before leaving the ROI we can plug (A.10) and (A.8) in (A.9), finally obtaining (3.4). □

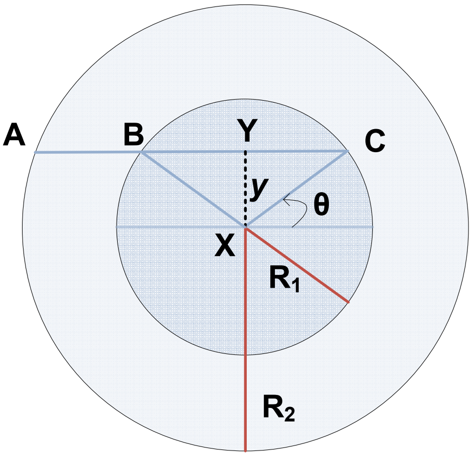


Figure A.2: Anchor zone and ROI.

Proof of Result 3: In order to prove Result 3, we first need to introduce the following lemma.

Lemma 6. *In equilibrium state, for a node traversing an AZ and a ROI, under the Random Direction mobility model, the probability of meeting k nodes before entering the ROI can be approximated as*

$$P_{BE}(met\ k) = \int_{R_2-R_1}^{\sqrt{R_2^2-R_1^2}} \left(f_{L_{AB}}(\ell) \frac{(2r\ell\lambda)^k e^{-2r\ell\lambda}}{k!} \right) d\ell \tag{A.13}$$

Proof. First of all, we compute the average length of the path that a node travels when it traverses the length AB (as shown in Fig. A.2). Since nodes travel in a straight line, the length L_{AB} of the chord AB is given by

$$L = |AB| = g_3(y) = \sqrt{R_2^2 - y^2} - \sqrt{R_1^2 - y^2} \tag{A.14}$$

The pdf for chord length AB is computed as with $\ell \in [R_2 - R_1,\ \sqrt{R_2^2 - R_1^2}]$.

If the trajectory of a node within the anchor zone is of length ℓ, the area swept is $2r\ell$, where r is the transmission range of each node, therefore using a similar approach as in (A.1), the probability of meeting k nodes along the chord AB is given by (A.13).

□

Proof. (Result 3) Using equations (A.13) and (A.8) in (A.9) we finally get (3.6). □

Appendix B

Proof of Theorem 2

Lemma 1 (Frequency of contacts)**.** *The frequency with which two nodes come in contact in the AZ, in the Poisson Jump mobility model, is given by*

$$\nu = \frac{2\mu r^2}{R^2} \tag{B.1}$$

Proof. Let us consider first the case in which the two nodes cannot jump out of the AZ. Assume first that one node is fixed, and the other jumps. As the location after the jump is uniformly distributed in the AZ, the chance of jumping within the transmission range of the fixed node is the ratio between the coverage area of a node, πr^2, and the area of the AZ, πR^2. Then the number of jumps needed for the jumping node to come in range of the fixed one is geometrically distributed, with mean $\frac{R^2}{r^2}$. As $1/\mu$ is the mean duration of the stopping time, the mean amount of time for the two nodes to meet is $\frac{R^2}{\mu r^2}$. The inverse of this quantity gives the frequency at which such an event takes place. If we consider now that both nodes jump, the frequency with which the two nodes meet is two times greater than the one obtained when one is fixed. □

In the Poisson Jumps mobility model, both nodes may exit the AZ after a jump. In an equilibrium state, the probability of a node jumping out is the same as a the probability of a new node jumping in, so that on average, the fact that nodes jumps in or out does not vary the mean rate of contacts between nodes in the AZ, with respect to the case of a closed system. Finally, when the system is in transient state (assuming the system starts from an empty state), the likelihood of a new node jumping into the AZ is higher than the likelihood of a node jumping out of the AZ, so that our computation of the frequency of contacts is conservative, and in any case it may be considered to be a valid approximation for the contact frequency over short time intervals.

Lemma 2 (Average availability)**.** *Consider an AZ in equilibrium state, with an average number of nodes equal to N, and let $\overline{n}$ and $\overline{m}$ denote the average number of nodes with and without content, respectively. Assume nodes move according to the Poisson Jumps model. Then $\overline{m} = \min\left(\frac{\mu(1-p)}{\nu}, \frac{\gamma}{\mu(1-p)}\right)$ and $\overline{n} = \frac{\gamma}{\mu(1-p)} - \overline{m}$, with ν given by $\frac{2r^2\mu}{R^2}$.*

Proof. The number of nodes in the AZ and the number of nodes with/without content are usually described as stochastic processes. Here instead we focus on the evolution of the mean of these system parameters over time, which can be modeled by the following differential equations:

$$\frac{dn(t)}{dt} = \nu n(t)m(t) - n(t)\mu(1-p) \tag{B.2}$$

$$\frac{dm(t)}{dt} = -\nu n(t)m(t) - m(t)\mu(1-p) + \gamma \tag{B.3}$$

The first term, in both equations, is derived as follows. Assuming the content to be uniformly distributed among nodes, the probability that a node has the content at time t is $\frac{n(t)}{N(t)}$. Then when two nodes come in range of each other, the probability that content is replicated is $2\frac{n(t)}{N(t)}\left(1 - \frac{n(t)}{N(t)}\right)$. In the AZ at time t there are $\frac{N(t)(N(t)-1)}{2}$ possible pairs of nodes. Thus, the average rate at which content is transferred in the AZ at time t is given by $\nu\frac{(N(t)-1)m(t)n(t)}{N(t)} \cong \nu n(t)m(t)$.

As for the second term in both differential equations, $n\mu$ gives the average jumping rate within the AZ. Since only $(1-p)$ of those jumps goes out of the AZ, the rate at which nodes with (resp. without) content jump out of the AZ is $n\mu(1-p)$ (resp. $m\mu(1-p)$). Finally, solving for the stationary case, and noting that, from Little's law, $N = \gamma\frac{E[T]}{1-p} = \gamma\frac{1}{\mu(1-p)} = \lambda\pi R^2$, we get $\overline{m} = \frac{\mu(1-p)}{\nu}$, with $\overline{n} = N - \overline{m}$. □

Proof. (Theorem 2) We assume border effects are negligible, which is a good approximation when $r << R$. The probability of getting the content after a jump can be written as

$$\begin{aligned} p_{jump} &= \sum_{j=1}^{\infty} P(j \text{ neighbors}) \cdot \\ &\cdot \quad [1 - P(0 \text{ out of } j \text{ neighbors have content})] \end{aligned}$$

where $P(j \text{ neighbors})$ is the probability of having j nodes in an area equal to πr^2. For the computation of $P(0 \text{ out of } j \text{ neighbors have content})$, we consider that in a stationary regime, and given that the location of each jump is uniformly distributed in the AZ, the probability for a node to have the content is $\frac{\overline{n}}{N}$, independently of its position in the

AZ. Even in a stationary regime, this is an approximation, as the replication mechanism favors the formation of small clusters of users with content. However, in the settings we consider, node densities are well below those which allow a strongly connected component to form, and $r << R$. In such conditions the connected components are small, so that this is a good approximation. Hence we have the following result:

$$p_{jump} = \sum_{j=1}^{\infty} \frac{(\lambda \pi r^2)^j}{j!} e^{-\lambda \pi r^2} \left[1 - \left(\frac{\overline{m}}{N} \right)^j \right] \tag{B.4}$$

$(1 - p_{jump})^{k+1}$ is the probability of not getting the content during a sojourn in the AZ in which a node jumped $k + 1$ times into the anchor zone (including the first jump with which the node entered the AZ). By summing over all possible numbers of jumps k of a user during its sojourn in the AZ, we get:

$$P = \sum_{k=0}^{\infty} p^k (1 - p) \left[1 - (1 - p_{jump})^{k+1} \right] \tag{B.5}$$

□