UNIVERSITÀ DEGLI STUDI DI MILANO

PhD School in
Computer Science

Computer Science Department

PhD in
Computer Science
XXVII° Cycle

From wide– to short–range communications: using human interactions to design new mobile systems and services

INF/01

PhD candidate:
Christian QUADRI

Advisor:
Dr. Dario MAGGIORINI
School Director:
Prof. Ernesto DAMIANI

Academic Year 2013/14

To my special Elisa...

Contents

Chapter 1

Introduction

The widespread diffusion of mobile devices has radically changed the way people interact with each other and with object of their daily life. In particular, modern mobile devices are equipped with multiple radio interfaces allowing users to interact at different spatial granularities according to the various radio technology they use.

The term *interaction* assumes different meanings depending on the spatial granularity we consider. The interactions mediated by mobile broadband network (3G/4G) allow user communicate by using the traditional channels – such as calls and text messages – and bring pervasivity to new communication media, such as Online Social Networks (OSNs) and Instant Messaging (IM), and contents sharing applications. Each user interaction uses network resources, in particular, the massive usage of the new communication media has led to a increasing demand of data traffic which has shown the limits of the actual mobile network architecture. Network operators are now looking for solutions to improve network capabilities and to make the overall network more flexible and scalable, in order to deal with future data traffic demand [21]. Radio technologies such as Wi-Fi and Bluetooth (BT) are enablers of medium/short range interactions. These interactions can be exploited to create mobile applications and services which allow users to discover and interact with the surrounding people and environment, such as local content diffusion services, proximity social discovery services, opportunistic contents sharing applications, and more. This medium/short range interactions can also be exploit to create Opportunistic Networks (ONs) which leverage human mobility to create contact opportunities. Proximity radio technologies, like Near Field Communication (NFC), are suitable to detect very short range interactions with objects of our daily life, thus allowing people to interact with them and to embed them in their social space according to the paradigms of the Internet of Things (IoT) and Smart Cities. These aforementioned kind of interactions are able to capture the willingness of mobile users to interact with other people or with nearby objects. For example, the aforementioned

technologies are able to emulate an intentional physical handshake between two persons as well as a willful touch to an object.

In this complex and rich scenario, many issues and challenges are still open from a technological, architectural, and mobile services and applications points of view. First of all, how to improve the scalability and flexibility of the cellular network in order to tackle the rising demand of data traffic. Second, how to guarantee a suitable Quality of Service (QoS) and Experience (QoE) to mobile users, which require more and more ubiquitous and integrated access to mobile services. Another issue concerns the management of wireless spectrum, which is a very limited resource. This has induced the research community to find solutions which use and integrate multiple wireless technologies to enlarge access bandwidth and to offload data traffic. In this field, many problems are still unresolved, for example; how to seamlessly integrate different alternative radio technologies with cellular network, how to scale up Wi-Fi/BT opportunistic network to very dense metropolitan area and guarantee a suitable QoS, or how to develop efficient offload network service both from users and operator point of view. Moreover, from the application and services design point of view, the rapid change in the habits of mobile users has led to a new kinds of applications and services which are closer to the needs of the users in mobility and they are able to exploit the capabilities of the modern mobile devices. For example, applications have included contextual research services based on the mobile user context (geographical position, personal interests, surrounding information, and more) which can be inferred by gathering information through sensors and radio interface (accelerometer, GPS, Wi-Fi, and more). Moreover, we can find sensing applications which collect data by using on-board sensors and send them to central services where they are analyzed to infer for example actual car traffic condition or road surface status. These types of mobile applications/services have posed many challenges, such as how to efficiently collect, to store and to analyze the huge amount of sensed data, how to identify mobile user context in order to offer to a user what she/he is really looking for, and how to preserve user's privacy.

The research community is progressively moving to heterogeneous network solutions which include many different wireless technologies seamlessly integrated to address a wide variety of use cases and requirements. In 5th-Generation (5G) of mobile network we can find multiple network typology such as device-to-device (D2D), vehicular networks, machine-to-machine(M2M), and more, which are integrated in the existing mobile-broadband technology such as LTE and its future evolutions [32].

In this thesis we consider, as a general scenario, people carrying multiple radio equipped mobile devises moving inside a metropolitan area served by heterogeneous and integrated radio access network. In this scenario, users can have experience all kinds of interactions described above. For example, they can perform traditional phone activities (calls and text messages), can access to multimedia contents through data connection, interact with the surrounding environment using Wi-Fi and NFC technologies. This can be useful to

retrieve information about city events, public transportation system schedule and delays local contextual advertisement as well as to interact with city objects such as museums, monuments, and so forth.

In this work we provide network solutions, mobile services, and applications consistent with the 5G mobile network vision by using users interactions as a common starting point. We focus on three different spatial granularities, long, medium/short, and micro mediated by cellular network, Wi-Fi, and NFC radio technologies, respectively. We deal with various kinds of issues and challenges according to the distinct spatial granularity we consider. We start with an user centric approach based on the analysis of the characteristics and the peculiarities of each kind of interaction. To accomplish this task we use multiple techniques. The set of used techniques includes knowledge extraction from datasets by means of data mining and statistical methods, network simulation as well as the analysis of the potentiality of a certain kind of interaction to create mobile services and applications. In the following sections we present a more detailed description about the contributions of this work.

1.1 Long range interactions using cellular network

The daily behavior and interactions of people carrying mobile devices are responsible for the traffic load of today's networks. In particular, cellular network is now suffering from an unprecedented demand of bandwidth and ubiquitous connectivity especially in metropolitan areas. Cisco forecasts that in 2018 the amount of mobile data traffic will be around 16 EB per month [21]. The actual mobile network architecture, characterized by inflexible and hierarchical structure, is unable to deal with this huge amount of traffic. Mobile operators are looking for solutions following three main directions. One is to enhance radio spectrum efficiency in order increase the bandwidth at radio access network. Another effort is to modify the network architecture according to new trends of Network Function Virtualization (NFV) and Software Defined Network (SDN), to improve the flexibility and the scalability of the overall network. Moreover, mobile operators are trying to integrate other wireless technologies, such as WiFi, to offload data traffic and to save radio resources.

The effort in this innovation process has involved many research communities including telecommunication companies and the 3GPP consortium – responsible for drawing up the standards of mobile network. With respect to the spectrum efficiency telecommunication company are looking for Heterogeneous Network (HetNet) solution [24, 9] consisting of a mix of macro-cells, low-power nodes (micro, pico, and femtocells), and also WiFi nodes. HetNet architecture offers better use of spectrum, reduced transmitting power consumption, as well as better radio coverage in densely populated areas. The adoption of HetNet architecture has lead to the development of new techniques of Self Organized Network (SON), which are well described in [4]. These techniques include automatic inter-cell interferences and traffic load balancing,

allowing Radio Access Network (RAN) to automatically manage network resources according to actual local context. The improvement of overall network flexibility and scalability is today a key challenge towards the definition of future cellular network. In this direction, telecommunication companies are now rethinking their core network moving from actual fixed and rigid hierarchical infrastructure to Cloud and distributed architectures [33, 12, 119, 59, 46] by using NFV and SDN approaches. Regarding network offload solutions we can find different approaches. Local IP Access (LIPA) and Selected Internet IP Traffic Offload (SIPTO) are standardized solutions in LTE rev.10 [1] which provide access to local resources (contents, devices, and so forth) directly from cellular network. LTE Direct (LTE-D) [95] is a new technology which use the same LTE licensed spectrum to provide a device-to-device (D2D) connectivity suitable for social and point of interests discovery services, as well as D2D communication channels. Other approaches concern the interoperability with other wireless technologies in particular Wi-Fi [5, 9, 47]. These approaches have led to an hybrid network architecture in which part of data traffic is delivered by means of opportunistic networks leveraging D2D connectivity.

All mentioned solutions are strongly related to mobile users behavior. Human mobility highly affect mobility management systems, HetNet deployment, and D2D connectivity. Human sociality impacts how people perform interactions by mean of mobile phones, and how they use different communication media. Users profiles determine accessed and shared contents providing useful information to cache or prefetch popular contents [100]. Thus, a clear comprehension of the different aspects of human behavior is a key point along the path of the creation of the future mobile networks.

The recent availability of large phone datasets [30, 49, 58, 16, 110, 86] containing information about locations, phone interactions, and Internet traffic of mobile subscribers has allowed research communities to study different aspects of mobile users behavior on a large scale. Mobility aspects have been studied in [45, 23, 88, 106] showing that the mobility patterns of people are quite regular and strictly related to users sociality and lead to the formation of spatial communities [18, 16]. Mobile data traffic was analyzed in [100, 110, 86, 99] characterizing which type of applications are causing the most of data traffic, the relationship between mobility and used applications, as well as the impact crowded events on cellular network traffic load. Human interactions mediated by mobile phone were studied both from temporal and social point of view. Temporal inhomogeneities of users phone activities have been studied in [10, 65, 58], showing that users have the propensity to organize their activities in bursts. Structure and dynamics of the underling social network have been studied in [66, 78, 118, 106, 83] by using the complex network theoretical framework. Studies on human phone interactions dynamics have led to the development of generative models [10, 63, 65] taking into account different aspects of mobile users behavior. These models could be used by mobile operator to simulate subscribers activities and generated traffic to better evaluate costs and benefits of new network technical solutions.

Contributions

In Chapter 2, we study different aspects of human behavior to support the development of a new network service and architecture, a new set of mobile services and applications. To perform this task, we deeply analyze a mobile phone dataset of a large Italian mobile operator by means of data mining and statistical tools. In particular, we provide the following contributions.

First we analyze users mobility by showing that humans exhibit regular pattern visiting a few preferred location. This result is in accordance with previous literature results [22, 45, 117, 116, 56, 85]. We further integrate mobility aspects by considering mobile phone call interactions between users who are in proximity each other (e.g. to update quickly when we are a few minutes late to a meeting, to gather a group of friends attending the same event, to share contents with co-located friends, and so forth). We found that this type of interactions are on average 15-20% of the overall call traffic. Based on these findings, we propose a new network architecture which leverages NFV/SDN paradigms. On this architecture we develop two mobile services able to offload part of data traffic from the core network. The first one is a contents delivery service leveraging the regularity of users mobility patterns to prefetch contents close to location from which it could be accessed by users. The second one is a network service which is aware of users proximity and is able to manage communication channels locally without involving core network resources. For the contents delivery service we provide a quantitative analysis of the benefits both from network operator and subscribers point of view, while for the other service we tackle its feasibility in the actual LTE/LTE-A standard architecture.

Second, we deeply analyze the temporal inhomogeneities of mobile phone interactions by proposing a novel theoretical framework for detecting and analyzing multidimensional bursts thanks to the introduction of a new burst detection algorithm and metrics suitable to describe multidimensionality features. In particular, we study the interplay between different communication media from users point of view by observing that users have the propensity to organize their phone activities in order to minimize switches between different media rather than follow some order induced by social importance. We argue that users tend to avoid time consuming tasks of changing media. Based on this finding we develop a multidimensional bursts generative model which is the first model taking into account the interplay between different communication media.

Last, we study how users manage theirs sociality by means of mobile phones and we find similar results as in [78] by observing that users heterogeneously distribute activities across their phone contacts. We extend this work by developing a simple method to automatically extract the most significant social ties from the analysis of calls log. Based on this result, we develop a mobile application helping users to manage their sociality proving a suitable

user interface to organize multidimensional phone interactions by minimizing the costs of media switching.

Chapter 2 is based on the following publications:

- S. Gaito, D. Maggiorini, C. Quadri, and G.P. Rossi. Selective offload and proactive caching of mobile data in lte-based urban networks. In *Mobile Data Management (MDM), 2013 IEEE 14th International Conference on*, volume 1, pages 271–274, June 2013
- Sabrina Gaito, Giovanni Manta, Christian Quadri, Gian Paolo Rossi, and Matteo Zignani. Groo-me: Handling the dynamics of our sociality on mobile phone. In *Wireless and Mobile Networking Conference (WMNC), 2014 7th IFIP*, pages 1–4. IEEE, 2014
- Christian Quadri, Matteo Zignani, Lorenzo Capra, Sabrina Gaito, and Gian Paolo Rossi. Multidimensional human dynamics in mobile phone communications. *PLoS ONE*, 9(7):e103183, 07 2014

1.2 Short range interactions in urban space

Wireless technologies, such as WiFi and Bluetooth, provide medium-short connectivity range and are today included in any mobile device on sale. By leveraging users mobility it is possible to exploit radio contacts among devices to build multi-hop ad hoc networks in which mobile devices, and consequently users, are active nodes of a network. Due to the nodes mobility and the limited radio range the connectivity is intermittent and each node needs to temporary store packets until the connectivity is reestablished. This aspect introduces a certain delay in packets delivery which is very high compared to other wireless networks – such as mobile phone networks and wireless mesh networks – made Opportunistic Networks (ONs) unsuitable for Instant Messaging (IM) services, multimedia streaming services or other services having strict time constraints.

Nevertheless, ONs can be used to support a wide range of delay tolerant service and application, for example cloud service synchronization (e.g., pictures/videos upload), mobile sensing (e.g., for traffic or road surface quality), local advertisement and contents diffusion, opportunistic collaborative games [75] and so forth. These kind of services and applications can be deployed on ONs and it is possible to offload the generated data traffic from cellular network.

Contributions

In Chapter 3 we explore the feasibility of ON deployed on Public Transportation System (PTS) which offers some benefits w.r.t. ONs made up by humans only. First, buses have no power constraints and can be equipped with powerful radio interfaces and high capacity storage appliances. Second, buses follows predefined paths and timetables which make mobility patterns more

predictable than humans ones. Last, PTS provides a good coverage across all metropolitan area.

We analyze the scalability of the Op-HOP routing algorithm [37] comparing it with the state of art of routing algorithms for PTS: MaxProp [14] and RAPID [7]. By means of extensive simulations on a real metropolitan PTS we show that Op-HOP outperforms both algorithms in terms of delivery time, packed delivery and resources usage. We then improve the performance of the entire backbone by introducing fixed nodes deployed at bus stops. We develop a suitable method for the deployment of fixed nodes. By using this method we show that a very limited number of fixed nodes are required to increase the network capacity and delivery ratio, as well as to reduce packet delivery time.

Chapter 3 is based on the following publications:

- Christian Quadri, Dario Maggiorini, Sabrina Gaito, and Gian Paolo Rossi. On the scalability of delay-tolerant routing protocols in urban environment. In *Wireless Days*, pages 1–6, 2011
- Sabrina Gaito, Dario Maggiorini, Christian Quadri, and Gian Paolo Rossi. On the impact of a road-side infrastructure for a dtn deployed on a public transportation system. In *Networking (2)*, pages 265–276, 2012

1.3 Micro range interactions and Internet of Things

The availability of micro range wireless technologies, as NFC, enables new set of mobile services and applications allowing users to interact with objects present in metropolitan area (e.g., museums, monuments, shops, and other points of interest), meeting the Internet of Things (IoTs) and Smart Cities paradigms.

Contributions

In Chapter 4 we exploit micro range interactions mediated by NFC radio technology to build up an online social network, named THINPLE. THINPLE is made of persons and things, and both can interact each other realizing the Internet of Things and People paradigm. In THINPLE the online social space of users reflects the offline users sociality which is expressed by person-to-person or persons-to-objects contacts. Links are created by means of NFC radio contacts only, this process of creation of social ties capture the willingness of users to create a relationship. As we can see this approach is totally different from traditional online social networks which are now suffering from proliferation of a very large number of worthless ties.

We develop the prototype of THINPLE and we use it to support a research study on difference between online and offline human sociality [40], and to support a mobile application to help tourists in keeping track of encountered monuments and point of interests.

Chapter 4 is based on the following publication:

- Sabrina Gaito, Christian Quadri, Gian Paolo Rossi, and Matteo Zignani. Thinple - the new online sociality is built on top of nfc-based contacts. In *Wireless Days*, pages 1–5, 2012

Chapter 2

Long range interactions in cellular network

In this chapter we present a comprehensive analysis of different aspects of mobile users long range interactions to design new network services and mobile applications. Thanks to the anonymized dataset from an important Italian mobile operator we are able to extract behavioral information about mobile subscriber and by leveraging the results of the analyses we propose new data traffic offload strategies, new mobile services and applications.

We start from the analysis of regularity of mobility pattern of mobile subscribers and we found that humans are regular in their mobility and have the propensity to visit few locations. Our results are in accordance with other previous literature works [22, 45, 117, 116, 56, 85]. Then we investigate the impact of phone call interactions between users who are close to each other. These interactions may occur for a quick update when we are a few minutes late for a meeting, among colleagues in a workplace, to gather a group of friends attending the same event or visiting the same place, to share contents with co-located friends, and so forth. We found that this kind of interactions are on average 15-20% of the overall call traffic, by exploiting this knowledge the mobile operators could benefit from proximity aware network architecture saving a significant amount of call traffic from core network.

The last aspect we analyze is the bursty behavior of humans phone activities. We start from consolidated literature works [10, 104, 65] and analyze the inter-event time distribution showing that it follows a power-law distribution. We extend previous works by considering time series composed by different types of phone activities (in our case calls and text messages). Each type of phone activities represent a different communication channel or dimension through which mobile user express her/his mobile sociality. The comprehension of the interplay between the dimensions is one of the key point in understanding the complexity of human behavior. In this direction we develop a new burst detection algorithm which is able to accurately detect multidimensional bursts given a time series of phone activities. We propose a theoretical framework capable to completely describe bursty behavior. First, we consider the burst at high level without investigating the internal multidimensionality by

defining three features: *persistence*, *focus*, and *frequency*. Persistence accounts for the persistence a user puts in doing a specific task. Focus accounts for the level of exclusive concentration a user is willing to assign to a burst activity avoiding interruption by different tasks. Frequency describes how frequently a user opts for doing a burst activity in the time scale of a day. Then we zoom in the inner structure of a single burst to analyze the interplay between the different dimensions by defining three features which completely describe the multidimensionality of bursts: *number of dimensions*, *symbol prevalence* and *interleaving degree*. Number of dimensions indicates the number of different types of phone activities performed user during burst activity. Symbol prevalence accounts for predominance of a communication media w.r.t. the others. Finally, interleaving degree quantifies how frequently user switches between different communication media. We found that users use both media fairly (calls and text messages) and the order with which users organize their phone activities does not seem to be influenced by social factor, but is mainly influenced by a media selection process that try to minimize the costly task of switching between different media.

After analyzing the different aspects of human behavior we exploit them to propose new mobile network services to help the network operators addressing the problem of continuous increase of demand for data traffic. In particular, we propose a network architecture based on Network Function Virtualization (NFV) and Software Defined Network (SDN) which is able to scale and to be deployed on demand in depending on traffic level and required Quality of Service (QoS). Based on this infrastructure, we present two network services. The first one is a proactive caching service which exploits the regularity of mobility patterns to prefetch contents close to the location from which users could require them. The second one is a proximity-aware service acting at edge network level; this service is able to detect and manage mobile phone calls happening between users who are in proximity to each other. Both these services allow mobile network operators to offload part of data traffic away from their core network, saving resources and potentially offering better QoS/QoE to mobile subscribers.

As already discussed, from the analysis of multidimensional bursty behavior we have learned that users organize their phone activities having the propensity to minimize switches between different communication channels. This counterintuitive fact lead us to design new mobile application – named Groo-Me – helping user to organized their phone activities by first selecting the person with who interact and then the communication channel. Moreover, the application maintains the users' personal network, helps users to control their core sociality (grooming coalition according to the Dunbar theory [28]), exploits burstiness and periodicity to predict the next user's activity, and collects statistics about the user's interactions on different channels.

In the last part of this chapter, we present a new multidimensional burst generative model which is able to correctly reproduce the characteristics of multidimensional bursts.

2.1 Related work

The recent availability of mobile phone datasets has given the possibility to study mobile users behavior on a large scale. Mobile phone dataset can be divided in two main categories based on type of data available.

The first category contains the CDR (Call Detail Record) based datasets collected by mobile operators for billing purposes. They usually contain information about mobile phone interactions (calls and text messages) providing the identifiers of users and their locations (based on the current serving cell tower). This kind of dataset can also contain Internet data traffic CDRs but they only indicate the amount of data traffic without any information related to the content. CDRs based datasets have some limitations. First, the location information are registered if and only if users perform phone activities, otherwise we have no information about users position. This pose some issues in reconstructing mobile users mobility pattern due to the holes in the mobility traces during the inactive periods. Second issue is the lack of details in Internet data traffic making impossible to know which kind of contents users accessed. Last issue is the coarse grain location information which is cell tower based, nevertheless this last limitation is considerably reduced in metropolitan area due to the high density cell tower deployment. This fact makes location information more precise. Examples of CDRs based dataset in [30, 49, 58, 16, 88].

The second category contains mobile datasets collected by network mobile operator for networking purposes and they contain detailed information about Internet traffic. In particular, they contain data traffic records of IP flows passing through the core network and control packets traffic. Due to the massive amount of data generated this kind of dataset are limited in covered time period, usually one day or less. Examples of this category are found in [110, 86, 99].

Mobility aspects have been studied in [45, 88, 16] where they have shown the regularity of mobile users mobility pattern. In particular, they found that tend to visit a few set of preferred locations and, by means of clustering techniques they identified home and work locations.

Social network structure has been studied in [80, 82]. These works have reconstructed the social network structure by exploiting mobile phone interactions (call and text message) and the locations of the users (by inferring the co-location of the mobile users). By applying the complex network theoretical framework to the reconstructed social network it has emerged that graphs properties are similar to those observed in the online social networks. Other works are more focused on the local properties of mobile social graphs. In [49] it was analyzed the communication properties of the links and it was introduced an indicator of their persistence. They have shown that persistent links are strictly related to reciprocal edges, meaning that communications in mobile phone social network are strongly bidirectional.

Temporal dynamics of human interactions through the diverse communication media have become a subject of intensive research because a clear grasp of the same is considered a key factor in understanding the formation of the today's information society. A vast and well-established literature regarding the properties of social networks built on each communication media is now available [72, 82, 84, 80, 77, 50, 31, 17, 105, 108, 62, 107, 71, 68]. All these studies have shown that a pronounced temporal inhomogeneity characterizes this type of communication activity, i.e. users perform sequences of rapidly occurring events, interleaved by long inactive periods. As a consequence, starting from the seminal work of Barabasi [10] who stressed the inappropriateness of the Poisson process in their modeling, human dynamics has become to be considered as bursty. Previous research on user communication temporal behavior has mainly focused on a sole communication activity. Nonetheless, human sociality is expressed through different communication channels - each channel describes a specific dimension of human sociality as a whole - and therefore the understanding of its dynamics and complexity may be improved by reckoning on all different dimensions together. Thus, the study of multidimensionality has become an inescapable fact when designing both practical and theoretical frameworks that describe human activities. A few seminal works [79, 102, 11, 13] have adopted a multidimensional approach to study the structural properties of social networks when multiple communication channels are considered, while [61, 64] model a collective bursty temporal process as composed of subprocesses and study spatiotemporal correlations inside utilization patterns of mobile service users.

The studies of the burstiness of human behavior have led to the development of burst generative models aiming to reproduce the characteristics of bursts. Starting from the work of Barabasi [10] the bursty human behavior is model by means of priority queue in which each event has a prefixed priority, the higher is the priority of a task the shorter is the waiting time of its execution. This model was extended in [63] by introducing variable priority tasks. A simple Markov chain model is presented in [65], which is designed to generate inter-event times. The model has two states which correspond to in-busrt (characterized by short inter-event time) and normal behavior (characterized by long inter-event time). The transition probabilities are derived from the bursts length distribution.

The studies on the data usage patterns was conducted mainly by the AT&T research group [60, 100, 110, 99]. Thanks to the availability of data traffic IP flows they analyzed how the mobile users use different mobile applications and where they access to different contents. In particular, they studied the interplay between the kind of used mobile application and the city area (e.g., if exists a difference between contents accessed in business area w.r.t. shopping one). Moreover, they studied the impact of the crowded events on the generated data traffic in terms of used mobile applications (e.g., browser, social media, and more) and accessed contents. These studies can help the

mobile network operators to better design and shape dedicated network architecture to improve quality of service, and to support crowded events.

2.2 Analysis of different aspects of human behavior

2.2.1 Datasets

CDRs based mobile dataset

We use a large anonymized dataset of mobile-phone Call Data Records (CDR) – containing call, text message activities and Internet access of mobile subscribers – gathered in Milano metropolitan area from March 26 to May 31, 2012, for a total of 67 days. The dataset contains records for more than 69 millions phone call records, 20 millions text message and more than 60 millions Internet accesses.

Each record contains the following information: date, hour, source user-ID and the cell-ID of the cell tower from which the phone activities was performed. In case of call and text message records it is also reported the user-ID of the receiver and the cell-ID of the cell tower in which the receiver was when she/he received the call or text message. Additionally, in case of phone call record it is specified the duration of the call in seconds. In case of Internet access records we have information on the amount of data uploaded and downloaded during the session. However we have not any information about the content of the data traffic for privacy reasons.

This dataset has two limitations. First the positions are relative to the current serving cell tower, coarser than GPS position, but in metropolitan area the deployment of cell towers is very dense made the location information more precise. Second, this is a billing dataset; meaning that we have location information if and only if the users perform any phone activities.

Other mobile dataset

We use a second mobile dataset which contains information of call, text message activities and Internet access in an aggregated form. The dataset covers the same metropolitan area from 1 to 31 December 2013. Unfortunately, this dataset is not CDRs based, so we have no access to the details of performed phone activities. Nevertheless, we have the per-location information about the amount of phone activities performed by mobile users. Moreover, we have an aggregated form of the spatial flow records of voice call. Each flow record contains the amount of voice call directed from one location to another over a time slice of 10 minutes.

In this dataset locations are square regions with sides 250 meters long and we have no information about the cell tower. However, the size of a single square region can be compared with the area covered by a single cell tower

in a metropolitan area. In the following we will use the term *cell* to refer to both cell tower and square areas.

Required preprocessing

In the following we present the result of the analyses on different aspects of the mobile users behavior. For each analysis we first preprocess the original dataset to extract a subset of users that is relevant for the specific purpose of the analysis. To help the reader, for each analysis we present the characteristics of the extracted subset of users in a dedicated sub section.

2.2.2 Mobility in urban space

Dataset preprocessing

The dataset preprocessing is performed as follows. We first limit the observation period to a single week, from 14 to 20 of May. Then we consider only those users *active* each day (a user is active when she/he generates at least one activity per day). Moreover, the selection applies to working days only, because mobility patterns and content requests are very different during the week-end. Finally, we select the users in the dataset to those visiting[1] at least two different cells every day because we are are not interested in users who remain all day in the same location[2].

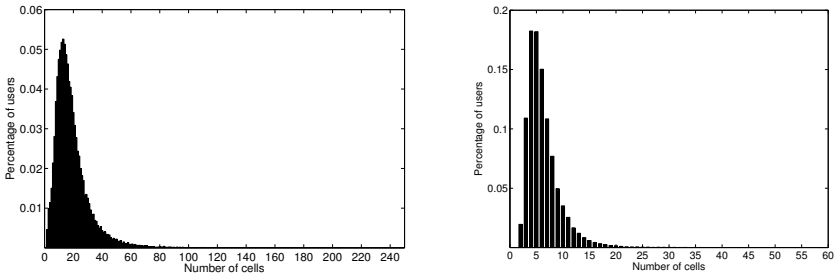
After the preprocessing we obtain a dataset of 49,067 users who were active over a total of 1,716 cells. Using this preprocessed dataset we extract all per-user activity patterns, where an activity pattern is defined as the set of cells where a given user is active.

Analysis

We initially consider the overall number of cells involved in the activity patterns of users during the considered week; for each user we compute the number of different cells she/he accesses during the whole first week and we show the related histogram in Fig 2.1a. The mean value of 19 cells and the median of 16 show that, apart from a tail of users accessing a very large number of cells, most have activity in a limited number of cells per week. This behavior is even more evident if the analysis is performed on a daily basis. In Fig. 2.1b we report the histogram of the mean (over the week) number of cell accessed by user in a day. In line with similar studies [22, 45, 117, 116, 56], most users are active in very few different locations/cells (mean: 6, median: 5), while a very small percentage of individuals show a higher mobility.

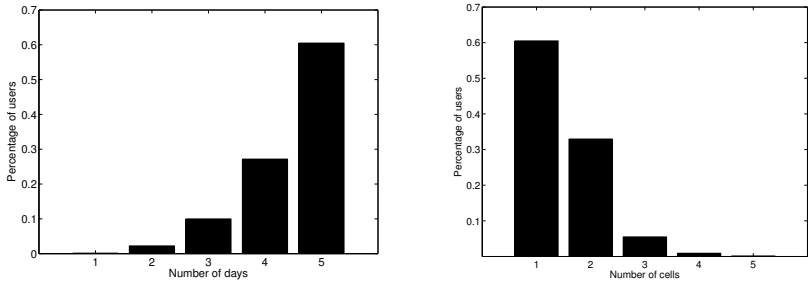
[1] In this context the terms *visited* and *accessed* are used to refer to the cell tower from which users performed their phone activities

[2] This affirmation is not completely true because users could have moved during the day without performing any phone activities.



(a) ***(b)***

Fig. 2.1: *Histogram of the number of cells accessed by users during the entire week (from 8:00 a.m. to 6:00 p.m.) (a). Histogram of the mean (over a week) number of cells each user accesses during a day (from 8:00 a.m. to 6:00 p.m.) (b).*



(a) ***(b)***

Fig. 2.2: *Histogram of the number of days users visit their favorite cell (a). Histogram of the number of cells required to serve users for all days of the week (b).*

To properly understand the regularity of the user mobility along the week is useful to know whether or not the set of cell it is changing daily. We extract this information by analyzing the per-user set of regularly accessed cells. The results are reported in Fig. 2.2a where we can observe that 60% of users have at least one favorite cell that they access daily. These results clearly state that individuals are accustomed to following regular mobility patterns that include a small number of cell visited daily. This information must be combined with the minimum number of cells required to serve a single user during the entire week. Fig. 2.2b shows that more than 90% of users require only two cells at most: 60% require one cell, the remaining 30% of users require only two cells.

2.2.3 In proximity person-to-person interactions

Mobile network allows users to communicate over long distance providing global connectivity, but our daily experience offers a rich record of interactions that we perform with people happening to be at short distance of our current location. In this section we analyze the impact of short distance interactions on the network of mobile operators.

Dataset preprocessing and methodology

We use two anonymized datasets containing voice call and text messages activities of mobile subscriber in the Milan's metropolitan area in two different time intervals.

The first dataset, *Data-Prox 1*, contains CDRs extracted from dataset described in Section 2.2.1 considering only call records over a time period of 9 week. By means of CDRs we are able to derive the spatial flow of phone activities, by identifying the cell in which the calling user is when she/he starts an activities and the one where the called user is when she/he receives the call or the text message. Clearly, this task is performed between users belonging to the same mobile operator.

As a second dataset, *Data-Prox 2*, we use the other dataset described in Section 2.2.1 extracting only the call flow records which are natively contained in the dataset.

We classify each spatial flow record in three different groups according to the destination cell: *in cell*, *in neighborhood*, and *other*. A flow is classified as *in cell* if the call or the group of calls have the same source and destination cell.

The notion of *neighborhood* is slightly different between the two datasets. In Data-Prox 1 the cells have a location-name attribute – street/square name or city zone – representing a coarse grain division of the metropolitan area. We consider a call to be in neighborhood if caller and callee cells have the same place name. In case of Data-Prox 2 we consider the cell's neighborhood as described in Fig. 2.3, in which the first Geo-Ring is formed by the cells adjacent to the calling location, the second Geo-Ring by those surrounding the cells of the first Geo-Ring, and so forth. If not otherwise specified, a cell is *in neighborhood* if it belongs to the first Geo-Ring starting from the calling location. Finally, a flow is classified as *other* if it does not belong to the other two categories.

In the sequel we will use the term *in proximity* to indicate the traffic belonging to the in cell and in neighborhood classes.

Impact of traffic in proximity on the overall network traffic

We firstly estimate the share of the global call interactions that can be ascribed to communications among individuals in the neighborhood. To this

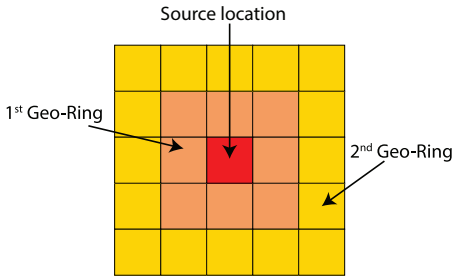
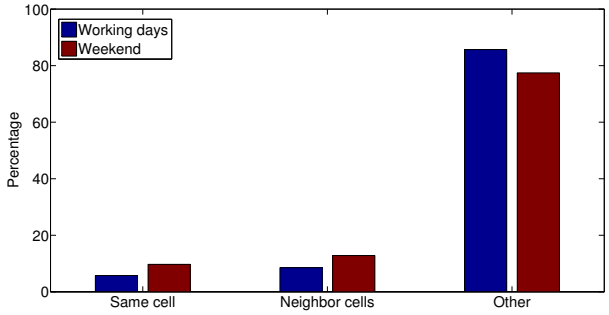
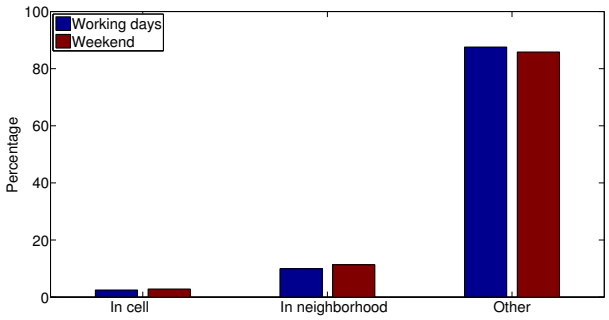


Fig. 2.3: *Example of Geo-Rings.*



(a) *Traffic distribution in Data-Prox1*



(b) *Traffic distribution in Data-Prox2*

Fig. 2.4: *Per-class traffic distribution in working and weekend days.*

purpose, we consider the aggregated traffic of the total amount of call flows. In Fig. 2.4 we report the traffic percentage for the three classes defined above in both datasets. We further separate working days from weekends to highlight possibly different human behavior. From the figure we can observe that nearly 12-15% of traffic during working days and 14-22% during the weekend is com-

posed by calls directed to in proximity cells. This observation has impact on the utilization of network resources and means that the proximity detection capability, when combined with the ability to switch this device-to-device *local* traffic at the edge of the core network, turns out to be a viable approach to remarkably decongest the core network. In practice, (north-south) traffic toward the P-GW would benefit of 12-15% reduction on average in favor of equal (east-west) traffic flows, if we limit operations to the inner circle of cells.

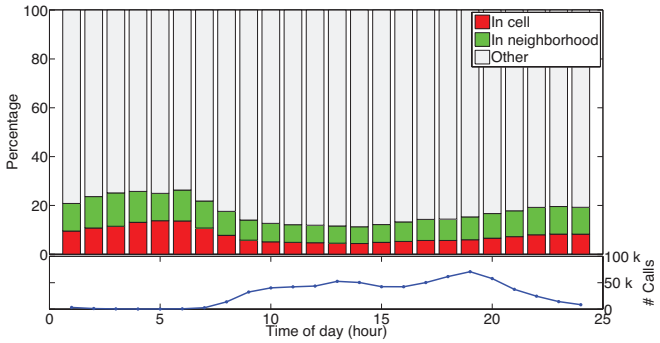
Fig. 2.5 reports the temporal distribution of in proximity traffic during the working days in the two datasets. We consider time slices of one hour and for each time slice we compute the percentage of calls that falls within the three proximity classes. For both datasets, in the lower part of the figures, we plot the overall call traffic (in number of calls) during the day. Calls are distributed according to a well known trend with two peaks, one at 11-12 a.m. and one at 5-7 p.m., and with most of call traffic concentrated from 8 a.m. to 8 p.m.. Of course, the impact of offloading in proximity traffic is more remarkable if performed during rush hours.

From Fig. 2.5a and 2.5b we can observe that, by considering only the most significant time slices, the percentage of in proximity traffic is around 15-20% and 15-16% for Data-Prox 1 and Data-Prox 2 respectively, confirming the results reported in Fig. 2.4.

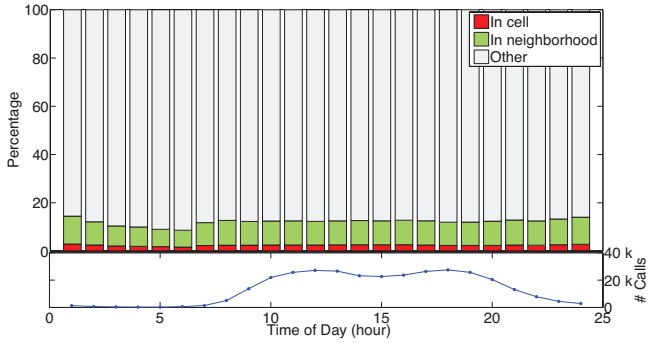
Spatial distribution of in proximity traffic

The question we answer in this section is the following: is the traffic in proximity fairly and uniformly distributed over all the cells, or a subset of cells shows different behavior? To show how in proximity call traffic is distributed over the considered geographic area, we firstly compute the per-cell percentage of calls directed in proximity. Fig. 2.6a and 2.6b plot the total number of per-cell outgoing calls against the percentage of in proximity calls. The two datasets show a slightly different behavior. In fact, the distribution of the outgoing calls in Data-Prox 2 (Fig. 2.6b) is concentrated in lower values (mean 852.8 and 95-percentile 3488) than those in Data-Prox 1 (mean 1386.1 and 95-percentile 3277), and the distribution of the percentage of in proximity calls has mean 10.40% and 6.28% for Data-Prox 1 and Data-Prox 2 respectively. Interestingly, in Fig. 2.6b we can observe that in both datasets cells producing higher volume of voice call traffic have also higher number of in proximity calls. It turns out that these cells are the best candidate taking advantage of a network infrastructure able to offload the local traffic from the operator core network. But how many cells should the mobile operator equip with offloading capabilities to achieve the desired level of core network decongestion?

To identify the trade off we sort the cells in descending order according to the absolute number of in proximity calls, and, for each specified offload level, we select the top ranked cells required to match the required offload level. The result is reported in Fig. 2.6c. We still observe some difference between the



(a) Data-Prox 1



(b) Data-Prox 2

Fig. 2.5: *Per-class temporal distribution of inter-personal traffic during a working day (top). Absolute number of calls during a working day (bottom).*

two datasets. In Data-Prox 1 the percentage of required cells increases more quickly than in Data-Prox 2 so that, to offload the 10% of traffic, the 50% and the 12% of cells is needed in Data-Prox 1 and Data-Prox 2, respectively. This behavior is explained by the above mentioned arguments. In fact, for dataset Data-Prox 2, the cells in the tail of both distributions, contribute significantly to the overall amount of offloaded traffic. By contrast, Data-Prox 1 does not present this characteristic, so it is required a greater percentage of cells to meet the same offloaded traffic level.

Finally, by using the GPS coordinates contained in the dataset Data-Prox 2, we are able to draw the heat map of the absolute number of in proximity calls. In Fig. 2.6d we report the map by labeling the most significant locations. As expected, they are highly populated city's areas, including the city center and the two main railway stations. The hottest location is placed in a business district, in between the two stations.

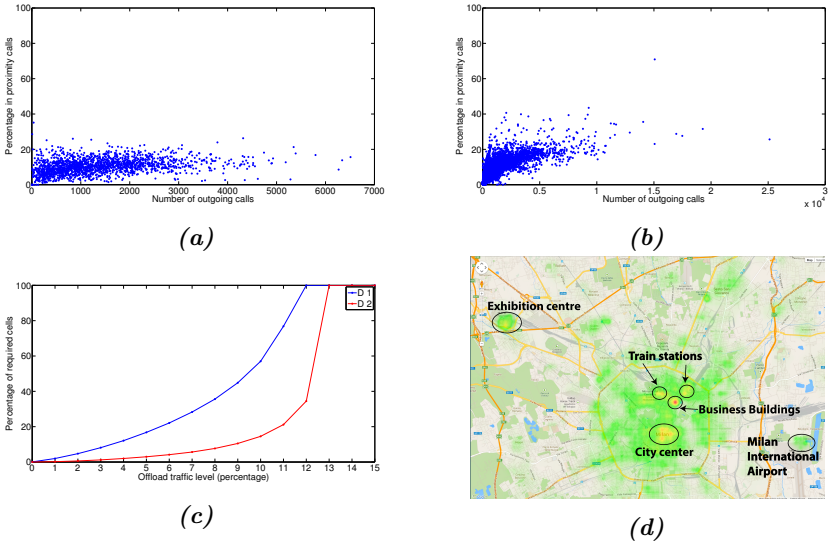


Fig. 2.6: *Correlation between the total number of outgoing calls from a given cell and the percentage of in proximity calls in Data-Prox 1 and Data-Prox 2 (a-b). Percentage of cells required to obtain a specified level of offloading (c). City of Milano: heat map of the absolute number of in proximity calls and the most significant places (d).*

Call duration analysis

In this section, we compare the call duration between calls classified as in proximity, and the others. Fig. 2.7 reports the cumulative distribution function of the call duration in minutes. It turns out that the duration of in proximity calls is shorter than other calls (mean 1.23 and 2.62 minutes, respectively), and that the percentage of missed calls, or rings, is slightly higher in proximity than otherwise (27% and 21%, respectively). These findings can be rooted to the well known human behavior of using such form of interaction to synchronize one another before a real-life encounter.

Impact of popular events

Mobile operators are used to deploy temporary cells in specific urban areas to absorb the extra access requests raising in conjunction with popular events [99]. To properly dimension and equip such an extra budget of cells, it is helpful to understand the impact of interactions in proximity during the event. To this purpose, we select the set of cells covering the area of two important events: the first is an important football match held at the San Siro stadium, while the second is the opening event at the Scala Theater. For the selected cells, we compare the amount of traffic in proximity the day of the event against

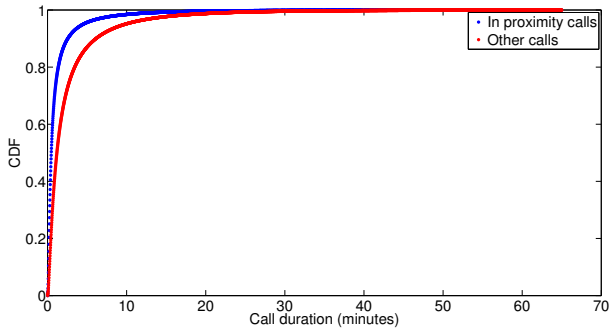


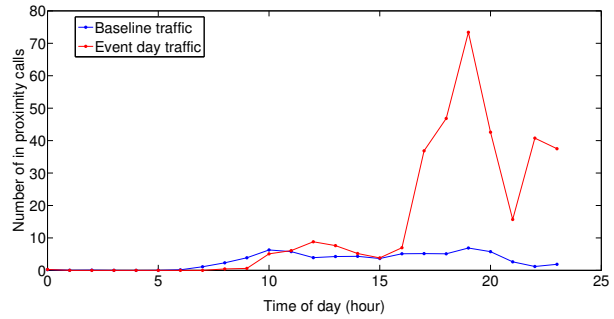
Fig. 2.7: *Cumulative distribution function of the call duration without considering missed calls and rings. Mean call duration is 1.23 and 2.62 minutes for in proximity/near and other calls respectively.*

the other days[3]. In Fig. 2.8 we only report the absolute and percentage value of the first event. We can observe that the football event has a significant impact in terms of increase in absolute number of calls. This is mainly due to the fact that the area surrounding the stadium is not densely populated. From the figure we can observe that during the event the percentage of in proximity call grows up to 150-200%. During the opening event we observe a lower increase of absolute number of calls in proximity, because the theater is located in the more populated city center. In this case we found that the percentage of in proximity call grows up to 50-100% w.r.t. the days without event.

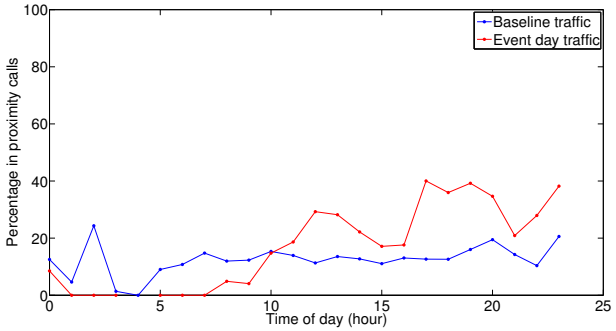
2.2.4 Multidimensional burstiness of mobile phone interactions

In this section we perform a multidimensional study of human sociality as an expression of the use of mobile phones, where the user has different communication media. In particular we focus on user temporal communication behavior in the interplay between the two complementary communication media, text messages and phone calls, that represent the bi-dimensional scenario of analysis. Our study provides a theoretical framework for detecting and analyzing multidimensional bursts by introducing a new burst detection algorithm and metrics suitable to describe multidimensionality features. In fact multidimensional bursts exhibit a complex inner structure that accounts for how individuals organize their activities once a burst is initiated. The interplay among the different dimensions can be fully described by defining the triplet of metrics $< d, \mathbf{p}, \iota >$ which captures the number of dimensions d, the dimensions prevalence $\mathbf{p}$ and the tendency to switch among dimensions ι.

[3] For the background traffic we average the amount of traffic over all other days.



(a) Absolute number of in proximity calls.



(b) Percentage of in proximity calls.

Fig. 2.8: *Impact of a popular football match event on the traffic level of in proximity calls. The considered event is the football match between Inter and Milan A.C., the two main football teams of Milan. The match lasted from 8:45 p.m to 10:30 p.m.*

The use of this general framework enabled us to offer empirical evidence of multidimensional bursty nature by analyzing the combined phone call/text message communication patterns of approximately one million people over a three-month period. Multidimensional analysis sheds light on human social interactions by phone and give answers to new emerging research questions: How do individuals schedule different phone activities? Is the mental selection process driven by technology or by social relationships? We find that the structure of the phone activity sequences is mostly influenced by the communication media. This behavior has to do with the uncomfortable and time consuming task of media switching which is able to condition the mental scheduling process.

Dataset preprocessing

The dataset was preprocessed to obtain significant time series of outgoing phone activities to highlight the active role in the initialization of the communication. The preprocessing phase focused on the dynamics of per-day activities of users. In fact, in line with the arguments proposed in [65], we are less interested in observing the role of human circadian and weekly activity patterns which act on both weekly and 24-hour time scales. In practice, we put more emphasis on human behavior in performing phone activities during a day than in natural life cycles.

The following analysis considers the outgoing phone activities of individuals as a sequence of discrete temporal events. For each pair (user, day), we build the time series of the outgoing phone activities performed by that particular user on that particular day. Finally, we obtain the set of time series representing our sample. Thus, an event time series s of a given pair (user, day) is defined as: $s_{user,day} = e(t_1, d_1), e(t_2, d_2), \ldots, e(t_m, d_m)$, where $e(t_i, d_i)$ is the $i-th$ event, an outgoing call or text message activity performed by $user$ at day, t_i represents the event's starting time and d_i is its duration, which we assume to be 0 in case of text message. From now on, we will use the terms time series and sequences as synonymous with s, disregarding indexes.

For this work purposes, we only considered the time series having a relevant number of both texts and calls, i.e. lying above a given threshold. This led us to obtain two sets of time series, namely *Data-Burst 1* and *Data-Burst 2*, by selecting two values of the threshold for outgoing calls and texts: for Data-Burst 1 the threshold is 25, while for Data-Burst 2 the threshold is 10. Moreover, in order to exclude anomalous users like robot-based event generators, telecom frauds, telephone sales, and such, we required that the daily activities should not exceed the threshold of 100 calls and 200 texts. After this preprocessing, we found Data-Burst 1 to contain 5,716 time series and Data-Burst 2 to contain 134,736. This way Data-Burst 1 accounts for time series expressing a very intense activity, while Data-Burst 2 weakens the activity level allowing us to check and generalize the behaviors observed in Data-Burst 1.

Burstiness of phone activities

It is commonly accepted that inhomogeneous time-dependencies within a sequence of discrete events show a heavy-tailed inter-event time distribution, where the inter-event time is the time elapsed between two consecutive events es. More precisely, in our case inter-event times are defined as $t_{i+1} - t_i$ for text messages and $t_{i+1} - (t_i + d_i)$ for call, where t_i is the start time of the event i ad d_i is the duration of call i.

To prove the burstiness of phone activities we apply the method proposed in Vasquez *et. al.* [104] and extended by McGlohon et. al. [76], Then, we fit the inter-event time data by using MLE (Maximum Likelihood Estimator),

and select the model with the minimum AIC (Akaike Information Criterion). Due to the considered temporal granularity of 1 minute, we take the discrete counterpart of Pareto distribution, the Zeta distribution with scale parameter α.

To properly capture the overall bursty nature dynamics of phone activities, we firstly compute the aggregated inter-event times distribution. The resulting complementary cumulative distribution function (CCDF) is shown in Fig. 2.9 along with the best fitting power-law and exponential distributions. It is visually evident and confirmed by the AIC criterion that the aggregated inter-times follow a power law distribution, accounting for inhomogeneous processes of call and text messages activities.

We further analyze the inter-event times distribution of the outgoing events on the single sequence of the datasets and we apply the same fitting methodology described above. In Table 2.1 we report the result of the fitting methodology on the single sequence, as we can see more that 75% of time series – considering all phone activities – fit with power-law distribution.

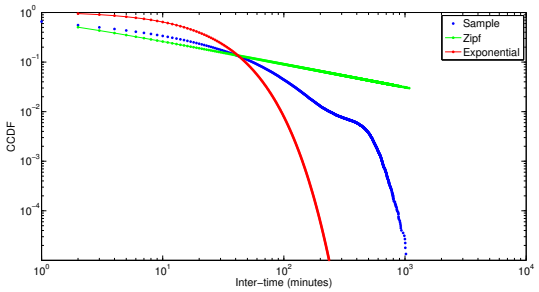
	Data-Burst 1		**Data-Burst 2**	
Media type	**Power law**	**Exp.**	**Power law**	**Exp.**
Call	78.83%	21.17%	78.12%	25.88%
Sms	73.69%	26.31%	54.40%	45.60%
All	87.16%	12.84%	76.67%	23.33%

Table 2.1: *Result of best fitting between power-law and exponential distributions.*

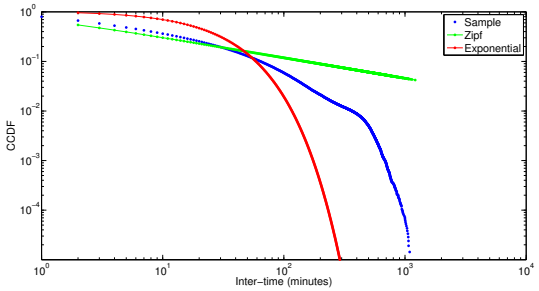
We finally focus on the similarity of bursty activity sequences by computing the distribution of the scale parameter α of the best fitting power law distribution computed above. The results are reported in Fig. 2.10. As shown in the figure, values are scattered around a peak at 1.5. This partially corroborates the results found in [104], which showed a single group of users with very similar behavior described by Gaussian distribution of α centered in 1.

Burst detection algorithms

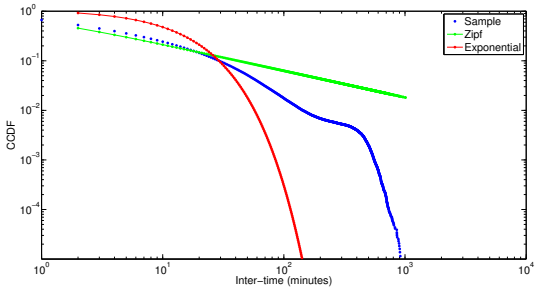
Given a time series it is important to correctly identify which events are performed in a burst and which are not. To this end different algorithms are developed to automatically find out bursts and they differ in the features they consider to identify the bursts: variation of event arrival rate [69], number of events occurred in a specific time window [114, 44] and inter-event time threshold [65, 58, 109]. In mobile phones dataset analysis the most used approach is the inter-event time threshold. This approach defines a burst as a group of consecutive events having inter-event time below a certain threshold.



(a) Call



(b) Text messages

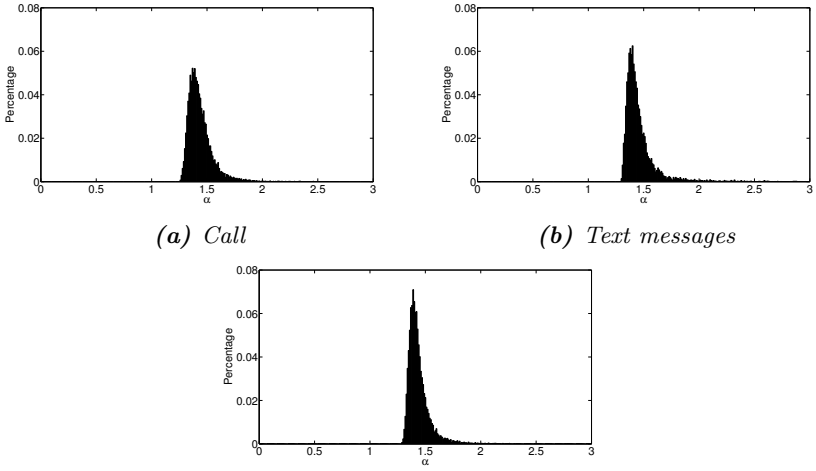


(c) All

Fig. 2.9: *CCDF of aggregated inter-event times along the best fit power-law and exponential distributions.*

Nevertheless, all these approaches are limited to one-dimensional case and do not consider temporal overlapping.

If we move from one dimension to a multidimensional burst time series the burst detection algorithm based on inter-event time of consecutive events is unable to correctly identify bursts, as they might be affected by a temporal overlapping between different dimensions. In fact, circumstances may exist where people communicate and socialize by different media simultaneously,



(a) Call *(b) Text messages*

(c) All

Fig. 2.10: *Distribution of α parameter of pairs user-day fitting with power-law distribution.*

as typing text messages while talking with friends nearby. Albeit rare, we observe this behavior in a few sequences of our datasets, where texts were interspersed with calls. Here we propose a different approach that is viable for any multidimensional case.

Let us start by providing a formal definition of burst. In a burst each event, except for the first and last ones, has at least k neighbor events within Δt. Let us consider the event time series $s = e(t_1, d_1), ..., e(t_m, d_m)$ and the function δ representing the time elapsed between two events $e(t_i, d_i)$ and $e(t_j, d_j)$ defined as follows:

$$\delta(e(t_i, d_i), e(t_j, d_j)) = \begin{cases} t_j - (t_i + d_i) \text{ if } t_j \geq t_i + d_i \\ \\ t_j - t_i \qquad\quad \text{if } t_i \leq t_j \leq t_i + d_i \end{cases} \tag{2.1}$$

For each event $e(t_i, d_i)$ of the sequence, we consider the set $E_i(\Delta t) = \{e(t_j, d_j) \mid \delta(e(t_i, d_i), e(t_j, d_j)) \leq \Delta t, i \neq j\}$, which represents the set of all events $e(t_j, d_j)$ having a time gap from $e(t_i, d_i)$ below or equals to Δt.

We define a *burst* as a sequence $b = e(t_s, d_s), \ldots, e(t_f, d_f)$ such that each set $E_i(\Delta t)$, with $s < i < f$, has cardinality greater or equals to k. According to this definition the burst length n, defined as $card(b)$, is $f - s + 1$.

From an algorithmic point of view, the given definition, and in general the idea of burst, can be reported in a way that fits with the model adopted by the density-based clustering algorithm framework [70]. A burst can be seen as a time period where the probability density function $\rho(t)$ exceeds a prefixed threshold. Density-based clustering is exactly a non parametric framework

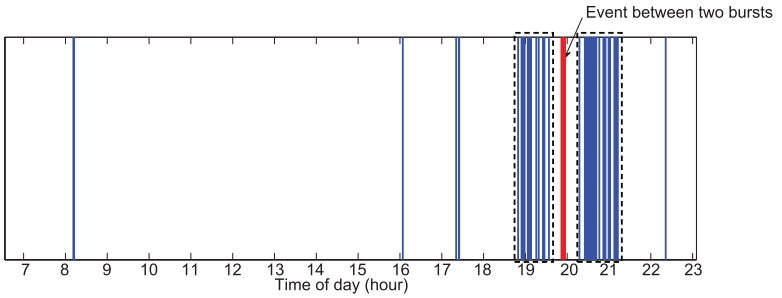


Fig. 2.11: *Example of event time series. This example explains how the burst detection algorithm works. The algorithm correctly identifies the densest regions (black dashed boxes) avoiding to include the event in between (red solid line).*

whose aim is the extraction of high density regions of $\rho(t)$. Therefore we assume that the points we group correspond to the time of the event e_i which comes from an unknown probability density distribution $\rho(t)$, while bursts are the clusters. Among the various density-based methods proposed in the literature, we select DBSCAN [34] because it scales to a large dataset and is robust against noise. Given a distance threshold Δt, and a threshold k on the number of the events within the interval $[t_i - \Delta t, t_i + d_i + \Delta t]$ (the choice of the interval $[t_i - \Delta t, t_i + d_i + \Delta t]$ in the construction of the neighborhood set accounts for the asymmetry of the function δ), the algorithm finds the maximally connected component (in terms of density reachability) of events at a distance smaller than Δt from some core points. By the term core point, we mean a point t_i such that the number of events in $[t_i - \Delta t, t_i + d_i + \Delta t]$ is greater than k. In Fig. 2.11 we illustrate how the algorithm operates on a toy sequence. In the figure we also observe an advantage that the density-based approach offers w.r.t. simple aggregation on the inter-event time [65]. Density-based methods overcome the so-called 'chaining-effect' which affects the single linkage methods as the inter-event aggregation. In fact the sole aggregation can result in different clusters merged by a 'chain' of single points between the clusters. In the figure the chaining-effect induces the two detected bursts to merge into one due to the single event in between.

Burstiness degree

We apply the above described detection algorithm to both datasets by varying the threshold Δt from 1 to 60 minutes and $k = 2$ (the latter value enables the comparison of our results with those obtained by implementing the single linkage methods). The results are in Fig. 2.12. In both datasets we found that phone activities are very bursty and, even when we execute the detection

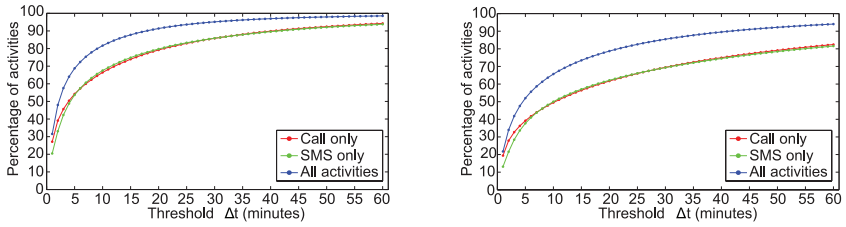


Fig. 2.12: *Mean of the percentage of events inside the bursts by varying the threshold Δt from 1 to 60 seconds: (a) Data-Burst 1, (b) Data-Burst 2.*

algorithm with the tight threshold value of 10 minutes, more than 80% of activities occur inside burst. In the same figure we also report the results we obtained by considering texts and calls separately. In both one-dimensional cases these percentages are much lower due to the fragmentation of bursts when the two media are alternated.

Burstiness, perceived as the generic concept of time inhomogeneity, has been widely proved for many human activities including phone activities. Such a behavioral concept has never been analyzed by means of quantitative measures that will favor an insight into a relevant human attitude as it occurs when considering burstiness of some physical and biological events, such as earthquakes and neuronal activities. With the notion of *burstiness degree* we want to characterize burstiness of human social activities providing a quantitative measure of the level of burstiness. In the following we define the three attributes which characterize a bursty behavior along with the metrics suitable to quantitatively describe them. The first two are properties of the single burst, while the third is ascribed to burst sequences. If we consider a time series s containing m bursts $b_1, \ldots, b_{mb}$, where the $u-th$ is composed by the events $b_u = e_u(t_s, d_s), \ldots, e_u(t_f, d_f)$, we can define the three attributes as follows:

- **Persistence**. This attribute accounts for the persistence a user puts in doing a specific task. The higher the number of activities within a single burst, the more the user is involved in her/his burst of activity. We can measure it by means of the burst cardinality metric, that is the number of events inside the burst: $persistence(b_u) = Card(b_u)$.
- **Focus**. This attribute accounts for the level of exclusive concentration a user is willing to assign to a single activity avoiding interruption by different tasks. The more the user is focused on it, the smaller the elapsed time between two consecutive events within a burst. We can measure it by computing the descriptive statistics (mean, median, and standard deviation) of the distribution of the intra-burst inter-event times, that it the elapsed time between the end of an event and the start of the following one inside

a burst. Formally, we define: $focus(b_u) = (t_{s+1} - (t_s + d_s), \ldots, t_f - (t_{f-1} + d_{f-1}))$.

- **Frequency**. This attribute describes how frequently a user opts for doing a given activity in the time scale of a day. We can measure it by compute the descriptive statistics of the distribution of the inter-burst inter-event times, i.e. the elapsed time between two consecutive bursts. Formally, given the sequence s of mb burst, we define: $frequency = \delta(e_u(t_f, d_f), e_{u+1}(t_s, d_s)), \forall u = 1, \ldots, mb - 1$.

A taxonomy of user bursty behavior naturally arises. For example, a user can be highly persistent in doing an activity by performing a relevant number of activities inside a burst, but he could have the attitude to multitasking, thus decreasing his focus; by contrast, we can imagine another user which is very focused but is able to perform a small number of activities. The overall burstiness degree is described by the triplet (burst-length, intra-burst inter-event times, inter-burst inter-event times), where the higher the three observed values, the higher the resulting burstiness degree.

The three burstiness degree components are computed on all sequences of bursts and characterized by means of their distributions and point statistics.

Persistence

High values for the metric $card(b)$ can be connected to the fact that a user decides to perform a relevant number of phone activities. The best fitting distribution of burst cardinality, $card(b)$, is definitely the tapered-Pareto distribution that perfectly matches with a cut-off behavior induced by the time threshold adopted in detecting bursts. In Fig. 2.13 the sample burst lengths and the tapered Pareto best fitting distributions are shown:

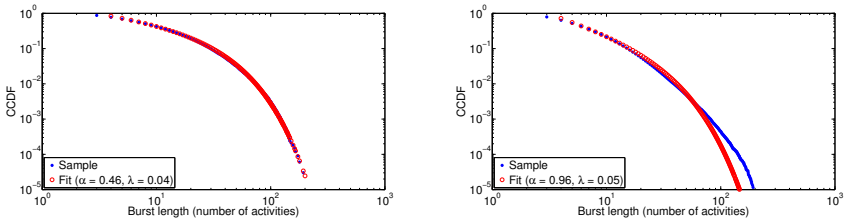
$$F(x) = 1 - \left(\frac{x_{min}}{x}\right)^{\alpha} \exp\left(\frac{x_{min} - x}{\lambda}\right) \tag{2.2}$$

where x_{min} is the minimum value of sample, while α and λ are the two distribution parameters.

The tapered-Pareto model well fits phone burst cardinality, accounting for the human decision process to dedicate themselves to a perform phone activities. The very high level of persistence of phone activities is underlined by the high mean and median values of the burst cardinality, as reported in Table 2.2.

Focus

The intra-burst inter-event times measures the average time to wait for the next activity to occur inside a burst. Its analysis is reveling of whether people have the attitude to dedicate time to a specific type of activity denying time for others or they have an attitude to task interleaving. The distribution of this index still exhibits the tapered Pareto model, as shown in Fig. 2.15, and

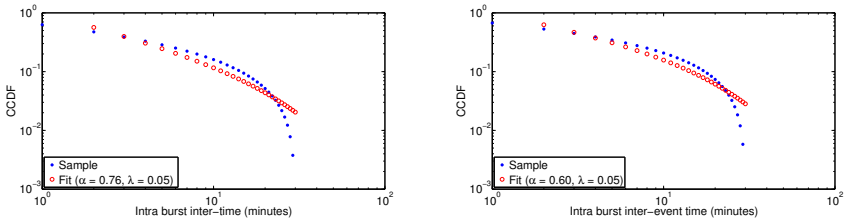


(a) *(b)*

Fig. 2.13: *CCDF of bursts length distribution and Tapered Pareto best fit. (a) Data-Burst 1. (b) Data-Burst 2.*

	Data-Burst 1			Data-Burst 2		
	Mean	**Med.**	**Std.**	**Mean**	**Med.**	**Std.**
Persistence (# events)	13.77	9.00	14.70	8.16	6.00	7.85
Focus (minutes)	4.71	2.00	6.45	6.21	3.00	7.16
Frequency (minutes)	107.3	64	121.1	141.9	92	142.5

Table 2.2: *Mean, Median, and Standard Deviation of the burstiness degree components.*



(a) *(b)*

Fig. 2.14: *CCDF of intra-bursts inter-times and Tapered Pareto best fit. (a) Data-Burst 1. (b) Data-Burst 2.*

the mean and median values, reported in Table 2.2, confirm that bursts are concentrated within a small time window. In fact, activities inside a burst are separated by less than five minutes on average, while it takes more than one hour between two consecutive bursts.

Thus people are highly focused on phone activities when they perform them.

Frequency

The analysis of inter-bursts inter-event times distribution shows the average burst generation frequency per a given type of communication.

In Fig. 2.15 we report the CCDF of inter-bursts inter-times and the relative best fitting distribution which still is the tapered Pareto distribution. Also in

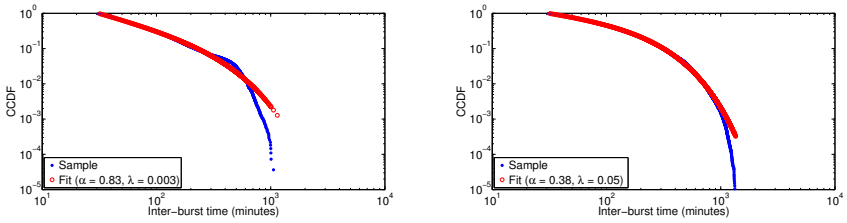


Fig. 2.15: *CCDF of inter-burst times and Tapered Pareto best fit. (a) Data-Burst 1. (b) Data-Burst 2.*

this case, the distribution parameters are very similar for both call and text message.

The statistics of inter-bursts inter-event times, reported in Table 2.2, show that bursts are significantly separated one another; in fact, both median and mean values are more than twice the 30 min. threshold we adopted in the burst detection algorithm. We can derive that phone activities are highly concentrated in specific time periods of the day.

In summary, the empirical evaluation of the defined indexes for burstiness degree allows to ascribe measurable data to phone activities, confirms that both phone calls and text messages can be rooted in a human scheduling process and provides an insight into the human behavior while using techno-media for their social interactions. In fact, it has turned out that:

- persistent - individuals are used to explicitly allocate the required time to phone activities, as revealed by high values of burst cardinality;
- focused - this time is almost exclusively dedicated and individuals have the tendency to approach social interactions with a single-task attitude, as shown by the very short intra-burst inter-event times;
- concentrated - phone activities are performed almost exclusively in dedicated time periods, as confirmed by the very long inter-burst inter-event times along with the very small number of phone activities performed outside bursts.

Multidimensional features

Here we focus on the characterization of multidimensional bursts as the very general burst category that includes one-dimensional bursts as the simplest case. In the following we omit the features that have been already defined in one-dimensional burst analysis and that are still valid in the general case (such as inter-times or burst length). Therefore we focus on a set of features, and corresponding quantifying metrics, able to fully characterize multidimensionality. Indeed, multidimensionality features each single burst, not the overall time series, and so models and metrics relative to burst sequence still hold.

Here we zoom in the single burst structure to describe its inner multidimensionality.

To this end, we first introduce the variable d that indicates how many different types of activity exist within a burst, *i.e.* the number of its dimensions, and denote as d-burst a d-dimensional burst. We can now represent a d-burst as a sequence of symbols belonging to the alphabet $\Sigma = \{0, 1, 2, \ldots, d-1\}$, where $card(\Sigma) = d$. In our bi-dimensional datasets, we code *text message* $= 1$ and *call* $= 0$ and represent bursts as binary sequences.

Secondly, we characterize the burst multidimensionality by considering the relative importance of the activities inside the burst. Given a d-burst, we define as *prevalence* the vector $\mathbf{p} = (p_1, p_2, \ldots, p_d)$ where p_i is the probability to draw the $i-th$ dimension symbol when considering a multinomial process on the burst:

$$p_i = \frac{\text{number of } i-th \text{ symbol occurrences}}{n} \tag{2.3}$$

where n is the burst length. In our case, $p_{text} = 1 - p_{call}$ holds and measures the *prevalence* of one of the two activities w.r.t. the other, accounting for the media selection preferences performed by a given user in a given burst of phone activity.

Nevertheless, these two features fail to describe how often a user switches from one medium to the other. Let us consider, for example text/call bursts. In addition to the one-dimensional burst, where the user decides to perform all activities on a single medium, multidimensional burst is a multiple symbol sequence. Symbols can be more or less interleaved inside a sequence, accounting for how often the user switches between media inside a burst. This way switches divide a burst into sub-sequences, each being a sequence of a sole symbol. An extreme case, very similar to the one-dimensional one, occurs when the burst can be divided in exactly two sub-sequences, one containing text messaging, the other call only. We name this burst a *disjoint* burst, as the user is definitely separating the two media. Single and disjoint bursts clearly account for a monotone behavior w.r.t selecting a particular type of activity; for example, a user may decide to use only one communication medium or to send all texts prior to performing other activities.

By contrast, bursts where symbols of different media are interleaved with one another are clearly an observable effect of the multidimensionality and can provide valuable insight into the selection process underlying the user's activities. For sane of clarity, we use the term *interleaved* to identify bursts that are neither disjoint nor one-dimensional. Of course, *interleaved* bursts exhibit different degrees of interleaving, which account for how often the user changes media or, equivalently, how many sub-sequences exist in the symbol sequence.

Intuitively, the higher the degree of interleaving, the farther the sequence moves away from the binary sequence representing the one-dimensional or disjoint burst. In the general case of a $d-burst$ the interleaving degree of a

burst is formally expressed as follows:

$$\iota = \frac{s + 1 - d}{n - d} \tag{2.4}$$

where s is the number of switches from one symbol to another inside the sequence. The numerator represents the difference between the number of sub-sequences inside the current burst, $s + 1$, and inside a generic disjoint type burst, d. The denominator is a normalization factor accounting for the difference between the number of events in the burst, n, and the minimum number of events of a d-dimensional burst, d. This coefficient assumes values in the interval $[0, 1]$: 0 means that the burst is a disjoint or one-dimensional burst, while 1 means that the type of activity changes from event to event. For example, let us consider the bursts having the events sequence in Table 2.3.

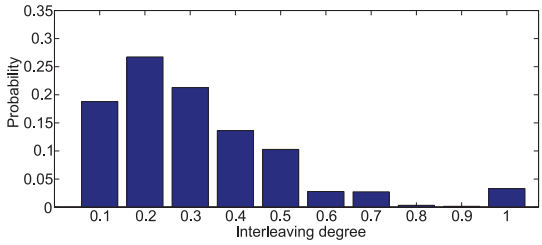
Burst	ι
0000011111	0
0011110001	0.25
0011001100	0.375
0101010000	0.625
0101010101	1

Table 2.3: *Example of burst sequences and the computed interleaving degree.*

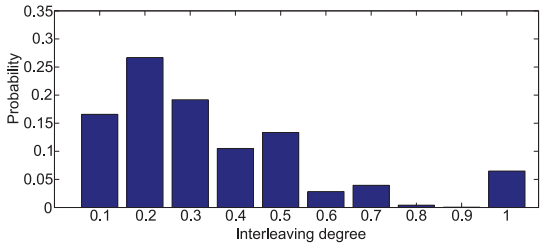
Finally, the triplet $< d, \mathbf{p}, \iota >$ fully describes the multidimensional nature of a burst.

We compute the interleaving degree on interleaved burst and we report the histogram in Fig. 2.16. We can observe that the overall degree of interleaving is very low, in fact more than 80% are below or equal to 0.5, accounting for a very low level of interleaving attitude.

As a final step of the multidimensional analysis, the comparison between the dataset's time series and a randomized dataset (null model) is the mechanism we adopt to see if the footprint of human behavior can be recognized in a user's sequence of events or if it is simply the consequence of a random selection of communication media. To produce the randomized dataset, we shuffle the user activities of the original dataset as follows: we start with the time series of activities and randomly permute the order of text messages and phone calls a user executes. This shuffling method allows us to leave untouched the inter-event times, ergo the detected bursts as well, while removing the activity type selection process, which in our case corresponds to the selection of communication media. The result of shuffling procedure is reported in Table 2.4. We can observe that the shuffled process causes a reduction of one-dimensional and disjoint burst types; in particular, we see a significant decrease in one-dimensional bursts. In fact, about 50% of bursts



(a)



(b)

Fig. 2.16: *Histogram of interleaved degree computed on interleaved burst. (a) Data-Burst 1: mean 0.32, median 0.29, and standard deviation 0.20. (b) Data-Burst 2: mean 0.36, median 0.31, and standard deviation 0.23.*

	Data-Burst 1			**Data-Burst 2**		
Burst type	**Original**	**Shuffled**	**Var. rate**	**Original**	**Shuffled**	**Var. rate**
One-dimensional	24.57%	6.42%	-73.86%	32.79%	10.72%	-67.29%
Disjoint	18.37%	14.49%	-21.12%	25.11%	21.51%	-14.33%
Interleaved	57.06%	79.08%	+38.60%	42.10%	67.77%	+60.95%

Table 2.4: *Comparative analysis between original and shuffled time series for each dataset. The shuffled time series was obtained from the original ones by performing a random permutation of the order of text messages and phone calls.*

are one- dimensional or disjoint, unlike what we would observe if the choice were obtained randomly.

The aforementioned arguments enforce the hypothesis that the execution order of phone activities is mainly affected by the need to minimize the switching overhead between different communication media (and the relevant apps). People experience a certain inertia which makes them to lean in single dimension and to persist in there up to completion of the planned burst activities.

These results magnify the burst nature of human communication and highlight the extent to which the multidimensional approach enriches the big picture of mobile phone communication.

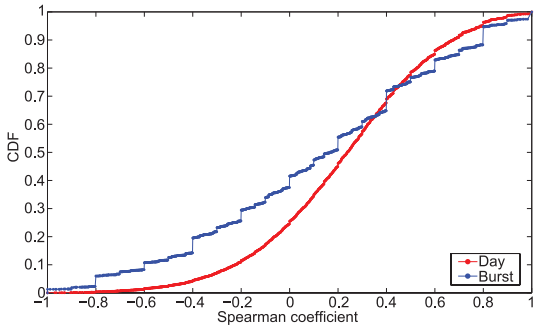


Fig. 2.17: **Distribution of Spearman correlation coefficient.** *CDF of the Spearman correlation coefficient computed on Data-Burts 2 considering sequences with more than three distinct receivers. We report the distribution for both day (mean 0.21, median 0.24 and std 0.36) and burst (mean 0.12, median 0.17 and std 0.50) level.*

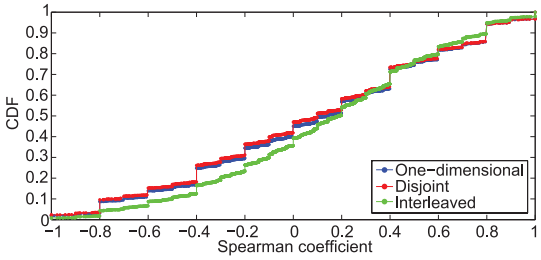


Fig. 2.18: **Distribution of Spearman correlation coefficient per burst type.** *CDF of the Spearman correlation coefficient computed on Data-Burts 2 by grouping burst by type and considering sequences with more than three distinct receivers.*

Multidimensional sociality using mobile phone

In the previous section we have shown that the processing sequences of human mobile phone activities are likely to be heavily influenced by media selection. This argument is slightly counterintuitive - we would expect human interactions to reflect the personal value (ranking) that individuals ascribe to their social relationships - and thus deserves further analysis. Provided that individuals have the attitude to organize their sequences of phone activities in multidimensional bursts with a very low interleaving degree, the question to answer is: does the order in which activities inside a multidimensional burst reflect the ranking that the user ascribes to her/his social relationships?

To provide a quantitative answer to this question, we perform a correlation analysis between burst activities and sociality rankings. The burst activities ranking has been performed as follows. We consider the sequence $(r_1, r_2, \ldots, r_n)$ where r_i represents the receiver of the i-th phone activity performed by user in a burst and we consider the vector given by the first occurrence of all the receivers. The burst rank of each receiver v is defined as the index of v in the vector of the first occurrences. For example consider the following sequence (b, a, a, b, a, c, c, b). The vector of the first occurrences is (b, a, c) which gives the following ranking $b = 1$, $a = 2$ and $c = 3$. We can stretch the scope of this ranking notion from a single burst to a full day by joining all the sequences corresponding to the bursts happened on a specific day.

We can describe the sociality ranking by organizing, in decreasing order, the neighbors of each user on the basis of the number of interactions the user has performed with her/his neighbor. Here we adopt the term 'interaction' to indicate a voice call or text message issued by a user. The sociality ranking has been shortened by removing neighbors not included in the burst/day ranking. This way the two rankings have the same cardinality. The evaluation of the Spearman correlation coefficient enables the comparative analysis of the two rankings.

We analyze the distribution of the Spearman coefficient ρ at both day and burst level and we report the relevant distributions in Fig. 2.17. To avoid artifacts induced by small number of points, we consider sequences with more than three different receivers. At a burst level we have nearly 28% of bursts showing high degree of correlation or anti-correlation ($|\rho| > 0.6$), while at day level we observe this percentage decreasing to 15%. These results highlight that the choice of the next activity is not driven by social importance. This still holds even in case of more regular behavior in terms of media selection, as it happens in one-dimentional or disjoint bursts. As we can see in Fig. 2.18, where the CDF of the Spearman coefficient grouped by burst type is reported, there is no significant difference between burst types. In fact, we observe that around 30% of one-dimensional and disjoint bursts show high degree of correlation or anti-correlation, which is similar to the the percentage observed in case of interleaved bursts, around 25%. We can conclude that the communication medium is the main ingredient to determine the organization of our sequence of activities inside a burst.

How do users distribute their social efforts using mobile phone?

People maintain a large number of relationships (*personal network*) but, in practice, their interactions are mostly concentrated on a small portion of neighbor nodes with strong ties (*grooming network* to recall the notion of Dunbar's grooming network [28]). We investigate how users distribute their social efforts, in terms of number of phone activities, among their networked neighbors. We use the *disparity* Y_i, previously used in [78] defined as follows:

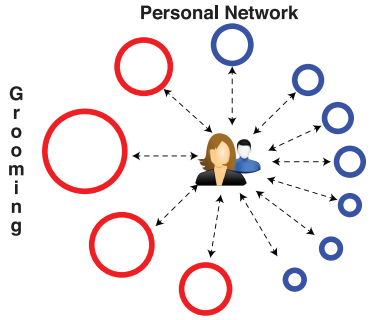


Fig. 2.19: *Schema of personal network of a single user. Each circle represent another user with which user have interacted. Blue circles are weak ties, while red ones are strong ties (users belong to grooming network).*

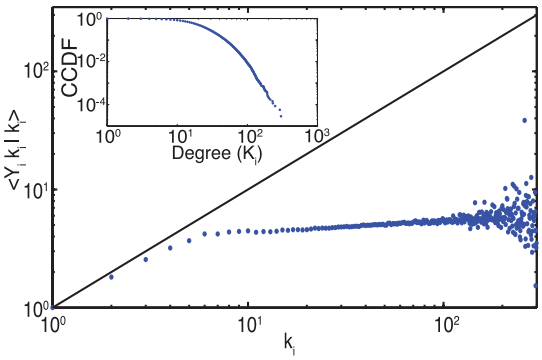


Fig. 2.20: $< Y_i k_i | k_i >$ *as a function of the degree* K_i *(blue points). The black line reports the limit case of a single person who polarizes the whole* w_i *(maximal heterogeneity). Inset: CCDF of the user's degree* K*, intended as the size of her/his personal network (users contacted by text messages or calls).*

$$Y_i = \sum_{j=1}^{k_i} \left(\frac{w_{ij}}{s_i} \right)^2 \tag{2.5}$$

where k_i is the degree of the user i, w_{ij} is the weight of link between the user i and one of her/his neighbor j, defined as the number of interactions (call and text messages) between i and j. The value s_i is the total number of interactions of user i, more formally $s_i = \sum_{j=1}^{k_i} w_{ij}$. The disparity is a measure of local heterogeneity and it is usually plotted as $k_i Y_i$ as a function of degree k_i. It assumes value 1 and does not depend on the k_i in case of perfect homogeneous behavior, while we have $k_i Y_i = k_i$ in case of severely heterogeneous behavior. In Fig. 2.20 we report the mean of disparity measure given the degree. We note that, except for low and high degree values (less then 10

Name	Analysis	Time period	Num. users	Record used
Data-Mob	Mobility	7 days	49,067	all
Data-Prox 1	In proximity phone call	9 week	734,149	call
Data-Prox 2	In proximity phone call	1 month	unknown	call
Data-Burst 1	Multi-dimensional burst	9 week	5,716 pair user-day	call and text messages
Data-Burst 2	Multi-dimensional burst	9 week	134,736 pair user-day	call and text messages

Table 2.5: *Characteristics of preprocessed dataset.*

and more than 100), the disparity value has a *quasi*-constant behavior. This result can be interpreted as a user propensity to enlarge their grooming network and to become more homogeneous by allocating her/his communication efforts to a greater number of personal contacts. To understand this behavior, we analytically derive a limit case in which a user allocates all efforts only to contacts inside his grooming network. Additionally, we assume user uniformly distributes efforts across the grooming network. Formally, we denote with G_i the size of the grooming network and we assign to each user belonging to the grooming network the same weight $w_{ij} = s_i/G_i$. We consider the following equalities chain:

$$k_i Y_i = k_i \sum_{j=1}^{G_i} \left(\frac{w_{ij}}{s_i} \right)^2 = k_i \sum_{j=1}^{G_i} \left(\frac{s_i}{G_i s_i} \right)^2 = k_i \frac{1}{G_i} = \gamma \tag{2.6}$$

where γ is a constant. We extract the size of grooming network given the degree as $G_i = k_i/\gamma$. Therefore we obtain that the size of grooming network is a percentage of the size of the personal network. In Fig. 2.20 we can observe that values span from 4.5 to 5.5 (for $10 \leq k_i \leq 100$) meaning that the grooming network includes around 20% of the entire personal network. These results can be interpreted as users propensity for enlarging their grooming coalition with increasing of the personal network size. By using these results the grooming network can be easily computed by taking the first quintile of the personal network.

2.2.5 Summary of preprocessed dataset characteristics

We summaries in Table 2.5 the name and the main characteristics of the preprocessed dataset presented in the previous sections. Hereafter, we will refer to a specific dataset by name.

2.3 Design of new network services for mobile operators

Mobile operators are today facing the challenges of a dramatically increasing demand on data traffic and are in urgent need to create the network capacity

required to accommodate the expected growth of traffic in the near future [21]. Operators are rolling out access network upgrades, such as the recent LTE. Nevertheless, simply increasing the access bandwidth cannot always be effective if it is not combined with complementary transformations of the core network, which is caught in the grips of exploding traffic flows which are extensively relying on a centralized, P-GW centric, architecture. These arguments are making the traffic offloading of the core network one of the primary goals that network operators must pursue today.

The mobile operators can use the data they collect about their subscribers to extract behavioral information to make offloading solutions more efficient both from the network and users point of view. From network point of view operators can optimize the utilization of resources, while from users point of view operators are able to guarantee a better Quality of Experience (QoE) to their subscribers.

In this section we propose two complementary services/functionalities able to offload part of mobile traffic from the core network. The first one is a proactive contents distribution service which is able to prefetch digital contents close to the location from which they could be requested by users. As a proof of concept, we focus on the distribution of newspapers and magazines available under subscription. Based on this assumption and by leveraging the regularity of users' mobility we provide a quantitative results of the performance of this kind of service from both operator and user points of view. The second service is an additional functionality provided at the edge of an operator network. This last functionalities – named Proximity-Aware Traffic Offloading Service (PATOS) – make the edge network aware of mobile interactions between users in proximity of each other. By leveraging SDN and NFV technologies, the edge network becomes able to create and manage users interactions in the nearby without involving the core network. We provide a description of our solution by showing its feasibility in the actual LTE network architecture.

2.3.1 Architecture overview

This section describes the NFV/SDN-enabled infrastructure supporting both services. It has been designed to meet the following general requirements:

- *User Equipment (UE) unawareness* – Proximity detection and traffic routing over local channels need to be transparent to the user and without involving her/his equipment. A seamless handover is also required whenever users enter/exit the local tracking area.
- *On-demand deployment* – We are not thinking of a stable deployment of these capabilities. The mobile operator should be able to activate or deactivate them when and where needed according to the dynamics of traffic load or to support planned popular events.
- *Flexibility and scalability* – In our case, flexibility is mainly required to ensure on-demand deployment. In fact, what we are considering here is an

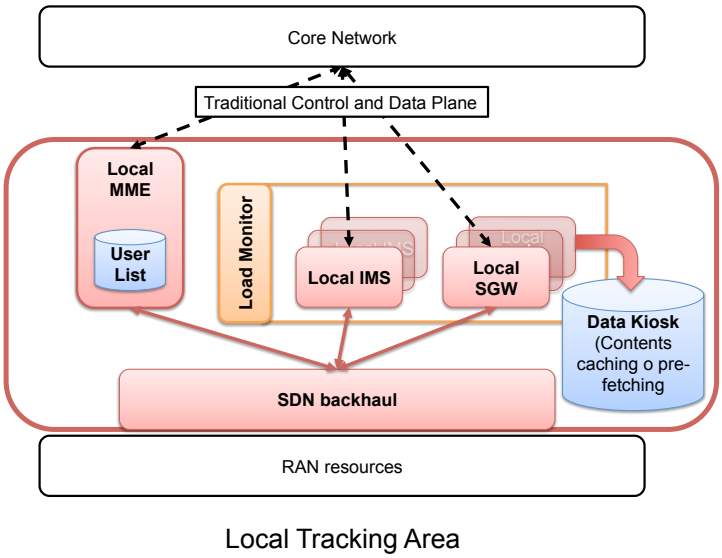


Fig. 2.21: *Traffic offload architecture overview.*

elastic concept of tracking area, where both size and functionalities may dynamically change over time. The mobile operator should be able to vary the number of eNBs belonging to it, and to flexibly allocate resources and functional modules whenever it decides to place proximity aware services to cover a given geographical area. Due to this relationship between tracking area a location-awareness, in the following we often use the prefix *local* to indicate functions and resources dedicated to the management of the traffic in vicinity.

- *Integration and interoperability* – To favor a quick service deployment, the PATOS-enabling proposal should be interwoven with current LTE Spec. and reuse LTE protocols and functionalities. At the same time, it should be easily placed inside the next generation of fully NFV/SDN-based cellular networks (see, for instance, [59]).

The aforementioned requirements cannot be satisfied by the current LTE Specifications. In fact, the LTE-Direct Specification [95] can only support interactions among devices in same radio range (∼500 m, line of sight), while the Local IP Access (LIPA) [94] approach has not the required flexibility to ensure on-demand service deployment, and similar arguments apply to the proposed LIPA's evolutions [94, 97]. For instance, the well-defined APN (Access Point Name) approach assumes that UEs support multiple APNs and are aware of local and non-local traffic. NAT-based solutions require deep packet inspection capability, which turns out to be a resource consuming and

poorly scalable task. Finally, by requiring UEs to use different radio bearers in case of local or non-local traffic, as in [19], a suitable policy to offload generic data traffic can be obtained but this is achieved at the cost of some scalability and transparency concerns. In fact, each bearer uses different Traffic Flow Templates (TFTs) to distinguish between local and non-local traffic, and TFTs need to be continuously updated whenever a UE enter or leaves the tracking area.

These arguments led us to propose a NFV/SDN-based architecture, shown in Fig. 2.21, which seizes on virtualization and programmability to satisfy the problem requirements, and insets the new functionalities in the current LTE Specification.

The *Radio Access Network (RAN) resources*, underlying the offloading architecture, consist of the set of eNBs in charge of providing radio connectivity to UEs attached to the given tracking area.

On top of RAN, the *SDN backhaul* is the virtual local area network providing connectivity amongst eNBs and the other functional modules involved in the tracking area operations. The *Local MME* acts as proxy of control messages. In particular, this module keeps track of mobile subscribers currently attached to the relevant tracking area, retrieves subscriber's attributes from Home Subscriber Server (HSS), and forwards to the core network the control messages not directed locally.

The *Local IMS* manages all local IP Multimedia Subsystem (IMS) messages and acts as a proxy for non-local IMS messages. The *Local SGW* (Local Serving Gateway) is the data plane proxy which enables the access to local contents. Both Local IMS and Local SGW cooperate with Local MME to perform the function of proximity detector and to instruct the SDN controller to create/modify/destroy control and data channels. Local SGW also provide access to *Data Kiosk* data storage which contains cached and pre-fetched contents that could be potentially accessed by users in the nearby area. Finally, the *Load Monitor* performs a real time traffic load monitoring and manages the resource utilization accordingly.

2.3.2 Mobile data traffic offload

The service we are going to present is a mobile data traffic offload service that helps mobile operators to selective reroute part of data traffic apart from core network. The data presented in [21] shows that mobile data contents are the best candidate to offloading because they are responsible for generating much of the mobile traffic growth through 2016 (with multimedia alone accounting for nearly 70 %). Moreover, digital contents can be available under subscription and are highly customizable, this two aspect can be exploited by mobile operator to offer a new kind of services by optimizing the contents distribution and at the same time by reducing the core network traffic.

This solution gives the opportunity to mobile operators to save core network resources and also provide better QoE to mobile subscribers. We present

our solution by steps. First we investigate the required number of cells to offer a push-based contents distribution service and to guarantee a suitable Quality of Service (QoS) to end users. Second we show the feasibility of a pull-based contents delivery service providing a quantitative analysis of the benefits by both network operators and users point of view. We use the distribution of newspapers and magazines as an example of pull-based contents delivery service able to prefetch digital contents close to the users. To provide the quantitative results we use the *Data-Mob* dataset.

Minimizing the number of cells

In this section we face the issue of finding the best selection of cells where to deploy/activate the offload infrastructure. By using the results of the analysis on the user mobility presented above the per-user preferred cells may be a good candidate for the deployment. In fact, by proactively caching the contents according to a per-user subscription, the operator would certainly be able to daily *push* the subscribed content to all users, thus ensuring that 100% of the offload data are efficiently routed outside the core network. Nonetheless, the general operator's achievement of the highest traffic breakout at the lowest cost can be pursued by attempting a cell optimization that selects cells shared among different users. To this purpose, we solve an instance of standard formulation of Set Covering applied to the first week of the dataset.

Our minimization problem is constrained to ensure coverage to all users for all days of the chosen week. The problem is defined as follows:

$$\min \sum_{c \in C} x_c \tag{2.7}$$

subject to:

$$\sum_{c \in C} a(u, c, d)\ x_c \geq 1 \qquad \forall u \in \mathit{Users}, \forall d \in \mathit{Days}, \tag{2.8}$$

$$x_c \in \{0, 1\} \tag{2.9}$$

where C is the set of cells accessed by users during the first week, $a(u, c, d)$ is a binary coefficient set to one iff the user u accessed to the cell c the day d, while x_c is a binary variable set to one if and only if the cell c belongs to the optimal solution. $Users$ is the set of all users, and $Days$ is the set of the five working days of the first week. The constraint of Eq. (2.8) ensures that each user is covered by at least one cell in each day of the week.

The number of cells resulting from the optimization problem is 1,081, the 63% of the total. In the following we will refer to this set of cells as C_o. This result is obtained after one hour of solver computation, with a relative gap from the lower bound less than 2.50%. Besides, the value of continuous relaxation is 1,031, about 61.1% of all cells.

Despite the strong constraint, we have obtained an encouraging result which allows the network operator to bound the deployment spending, while covering all users. We evaluated this optimal placement with the dataset of the second week, still considering a content *push* approach. By leveraging the strong regularity that users show in their activity patterns, the placed infrastructure ensures more than 95% of successful offload of the considered traffic.

A Pull Approach to Content Delivery

The *push* approach we considered so far is highly effective but slightly operator centric. In fact, it assumes that users accept a delivery service that autonomously pushes contents to them when they happen to transit under coverage of the cell selected by the operator. This can be effective in practice for a subset of contents but not for their totality. Many users may actually be willing to freely access their contents anytime and anywhere by *pulling* them from the Internet. In the following, we consider such a *pull* approach and we evaluate whether the described optimal placement remains effective under changed access conditions. To evaluate this new setting, we randomly select, for each user, 40% of her/his daily cell accesses and we use them to describe the users' requests for digital contents.

As a first performance index, we evaluate the amount of requests we can serve through the offload infrastructure and compare them with those served through the operator's core network. In Fig. 2.22a we report the traffic (aggregated in bins of 10 minutes) along a sample day, with all days behaving quite similarly. The black bars report the number of offloaded requests, while the grey bars report the number of requests served by the core network. The sum of both bars is the total number of users' requests. While the overall number of requests grows along the morning, has a peak in the launch pause and then remains quite constant during the afternoon, the percentage of offloaded traffic remains stable, as shown in Fig. 2.22b. In fact, the traffic breakout remains quite stable, around 80% of the total. This is a good result from the operator's perspective, for it shows the viability of the placement under different settings of access distribution.

From a user perspective, we assume that a mobile user may experience a somehow improved service efficiency when exploiting the offload platform. This argument makes interesting to estimate how fairly this opportunity is distributed among the set of users. Such a measure highly depends on the considered dataset, however, under our setting, the percentage of requests offloaded in a representative day is reported in Fig. 2.23a showing that about 50% of the users have all requests served by the offload infrastructure, while only few users experience less than 50% of requests offloaded.

We then enlarge the analysis window to the whole second week and report the results in Fig. 2.23b. On a weekly basis, the performance are very satisfy-

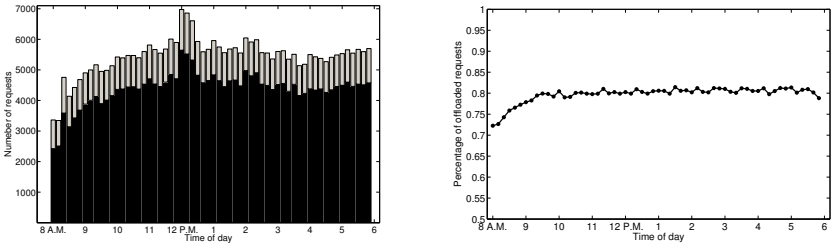


Fig. 2.22: *(a) Network traffic load from 8:00 a.m. to 6:00 p.m. of a representative day. Each bar covers an interval of 10 minutes. Black bars show the traffic load of offload network, grey bars show the traffic load of core network. The sum of the two bars show the traffic load of entire network. (b) Percentage of offloaded requests.*

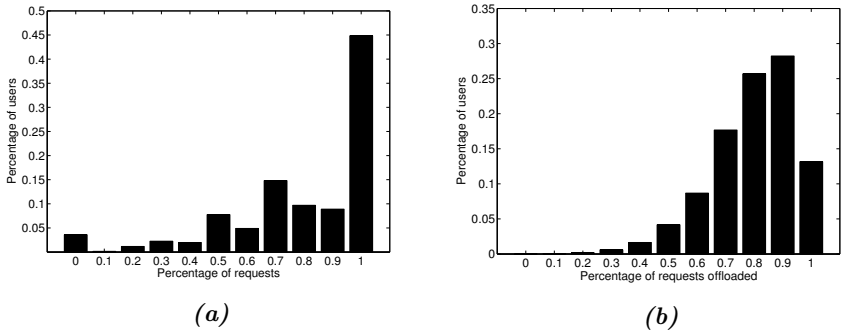


Fig. 2.23: *Histogram of users' percentages of offloaded requests in a single day (a). Histogram of users' percentages of offloaded requests over the entire week (b).*

ing; overall, 95% of users can experience improved performance in more than half of their requests when compared to core-routed traffic.

Summarizing, we can say that from the network operators perspective the results are encouraging, allowing them to save about 80% of requests under subscription. From users perspective, the daily analysis showed a good performance; for 90% of users the offload infrastructure satisfies at least half of user requests, and 45% of users are always satisfied by infrastructure.

The Placement of Data Kiosks

As described in Fig. 2.21, we assume that the offload platform includes data kiosks which we envision to exploit for enhancing mobility support in a news delivery system. In particular, we consider the delivery of two similar kinds of

data contents: newspapers and weekly magazines. Both contents are characterized by a sort of *durable subscription* [103] where kiosks cache the subscribed contents, independently of whether the user is connected to the associated cell or not, for a specified time of content validity (one day and one week, respectively). For the purposes of this thesis, we do not further differentiate subscriptions inside each content category and assume that newspapers are mostly accessed in the time window from 8:00 a.m. to 11:00 a.m., while magazines have a larger access time window, from 8:00 a.m. to 18:00 p.m.. In this caching system, contents are replaced daily or weekly, according to the class they belong to. We want to define a cell selection process to proactively cache contents based on the knowledge of the regularity of activity patterns of subscribers. The problem to solve in this case is slightly different w.r.t. the aforementioned one: here, we use the first week in the dataset to identify the per-user subset of most visited cells and we place the subscribed content in the associated kiosk. In practice, we leverage the regularity of user mobility to increase the probability of finding the cached content when the user will generate her/his content request during the second week.

The resulting new matching function $f_u(c)$ to each cell and for each user u is as follows:

$$f_u(c) = e^{d_u(c)} \sum_{d \in Days} w_u(c, d) \tag{2.10}$$

where c represents a cell, d_u is the number of days in which the user u accesses cell c, $w(c, d)$ is the number of accesses to cell c during the day d and $Days$ is the set of trial days. This function allows us to consider, for every cell and for every user, both the number of accesses in a day, $w_u(c, d)$, and the number of days it has been accessed by the user, $e^{d_u(c)}$. For each user u, cells are ordered by descending matching values and the first n elements are selected as favorite cells. Thus, we denote with C_u the set of the favorite cells of user u. In the following analysis we consider $n = 3$, being other cells not relevant according to Fig.2.2b. In the optimization process only the subset of favorite cells belonging to the offload infrastructure ($C_u \cap C_o$) is considered. If the intersection between C_u and C_o is empty, then the user is discarded because it will not benefit from the offloading infrastructure anyway. Thus, the set of cells where minimization will take place is C_f defined as:

$$C_f = C_o \cap \{\cup_{u \in Users} C_u\} \tag{2.11}$$

As above, we find the optimal set of cells for the whole set of users by solving an instance of the set covering problem defined as follows:

$$\min \sum_{c \in C_f} x_c \tag{2.12}$$

subject to:

$$\sum_{c \in C_u \cap C_f} x_c \geq 1 \qquad \forall u \in \text{Users} \tag{2.13}$$

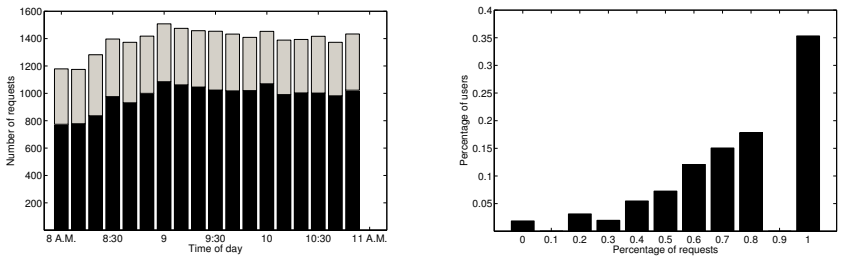


Fig. 2.24: *Network traffic load from 8:00 a.m. to 11:00 a.m. of a representative day (a). Histogram of users' requests offloading percentage for newspaper subscription (b).*

and

$$x_c \in \{0, 1\} \tag{2.14}$$

where x_c is a binary variable set to one iff the cell c belongs to the optimal solution.

Optimization is performed using mobility data of the first week while network access simulation is performed using the second week. User content retrieval is simulated by randomly selecting one cell access per day in the time window planned by the service. Since we suppose that each user submits only one request per type of service during a day, two independent selections are performed for each user.

Newspaper Subscription Performance Evaluation

After optimization, we are left with 993 cells where caching is required (92% of total infrastructure cells and 58% of the total cells). This scant reduction is due to the limited number of cells accessed during the morning. As a matter of fact, the ratio between the number of cells accessed in the morning and those accessed during the whole day is 0.88, meaning that most cells are accessed later in the day.

By analyzing the performance from a network operator point of view, we obtain the results depicted in Fig. 2.24a. In the figure, we report the traffic (aggregated in bins of 10 minutes) along one representative day, being all days very similar. Black bars show the traffic load of offload network while grey bars show the traffic load of core network; the sum of the two bars is the traffic offered to the entire network. The percentage of traffic offloaded is, in the average, 70% of the total traffic.

From the user perspective, the best download performance are experienced when the content is accessed by means of data kiosk. To evaluate these performance, we compute for each user the percentage of requests satisfied in

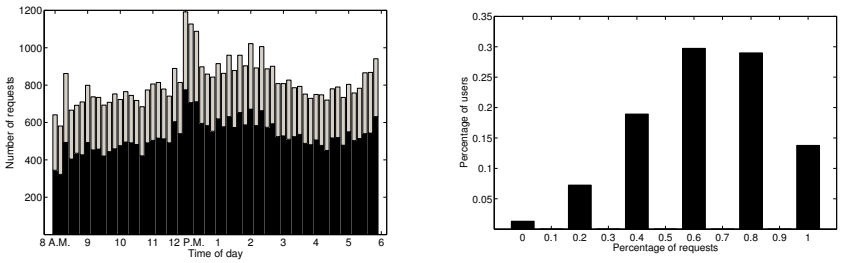


Fig. 2.25: *Network traffic load from 8:00 a.m. to 6:00 p.m. of a representative day (a). Mean user offloading percentage for magazine subscription (b).*

offload. The related histogram on all users is reported in Fig. 2.24b. As we can see, 35% of users are always satisfied by the offload infrastructure, while more than 70% of users are satisfied more than half of the times.

Magazine Subscription Performance Evaluation

By extending the time window to an entire day (from 8 a.m. to 6 p.m.) we observe a stronger reduction in the number of required cells. In fact, the number of caches to be deployed drops from 993 to 779: 72% of total infrastructure cells and the 45% of the total cells.

Following the same methodology as before, Fig. 2.25a reports the overall percentage of traffic offloaded along a day. Also in this case, the percentage of offloaded traffic is quite constant along the day, with a mean percentage value of 63%. The performance index is lower than the newspaper service running in the morning since most of the users access a larger set of cells and, as a consequence, the probability to request for a content in a cell without cache is greater. As before, the behavior for all other days is very similar.

Performance from users perspective is reported in Fig. 2.25b. Also in this case we can observe that more than 70% of the users have a percentage greater than 0.5, meaning that more than half of the requests are performed using the offload infrastructure. Nevertheless, the percentage of users who get always offloaded is lower than morning service: below 15% of the total population.

2.3.3 Proximity-aware person-to-person interaction offloading

The analysis conducted on the datasets *Data-Prox 1* and *Data-Prox 2* has showed that more than 15% (in same case more than 20%) of mobile phone calls involve persons who are in proximity each other. This fact confirms that mobile operators and users could both benefit from proximity awareness because it would allow better network resources utilization and improved user's

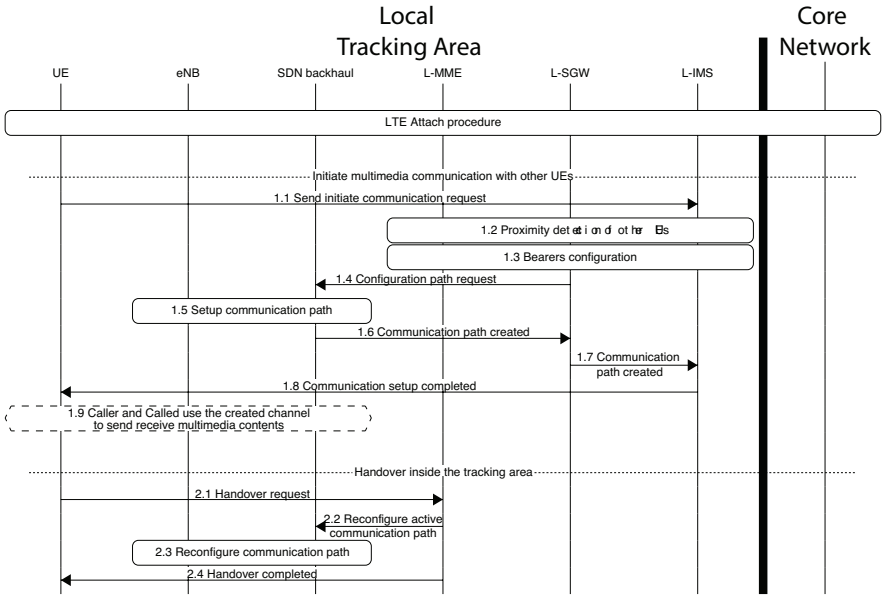


Fig. 2.26: *Message sequence of the establishment of a multimedia communication channel between two mobile subscribers*

QoE. This section describes the set of simple functionalities that should be deployed at the edge of the core network to deploy a Proximity-Aware Traffic Offloading Service, or PATOS. PATOS is composed of the following two components:

(i) **proximity detector**: it detects whether the called person happens to be below a specified distance from the calling current location (for instance, inside the workplace, the business district or a few blocks away from here).

(ii) **device-to-device communication**: this component establishes the device-to-device channel between calling and called devices. This channel is operated outside the core network whenever the devices are detected to be in proximity.

PATOS message flow

The PATOS message flow is described in Fig. 2.26. The procedure has been designed to ensure transparency to the user and to leave unchanged current LTE operations. As a result, the attach procedure is the standard LTE attach procedure without any additional dedicated bearers or well-defined APNs configuration.

When a UE is willing to initiate a IMS session, it sends an *initiate communication request* (1.1) to local IMS server specifying the phone number of called UE. Local IMS cooperates with local MME and local SGW to perform

the *proximity detection* task (1.2) and, if the callee UE is actually in proximity, the *bearer configuration* (1.3) procedure is started. This procedure is needed to guarantee the seamless handover when UEs leave the tracking area by maintaining the information of currently active communication channels between UEs inside the tracking area. As a next step (1.4), L-SGW sends a *configuration path request* to SDN backhaul to setup communication path between the eNBs. Once a path is created, the SDN backhaul notifies L-SGW which notifies the L-IMS that path is created and it is ready to be used (1.6 and 1.7). At this stage L-IMS replies to UE (1.8) and communication can start.

UEs are supposed to move during an IMS session. When a UE leaves its current tracking area, the standard LTE handover procedure is adopted; should the UE still remain in, the L-MME sends a *Reconfigure active communication path/s* request (2.2) to SDN backhaul to reconfigure the communication path according to current location of UEs (involved eNBs).

2.4 Design new mobile application to manage multidimensional human interactions

We have seen that mobile phone activities are usually organized in bursts showing the temporal inhomogeneity of the humans behavior in performing social interactions. We have also seen that users use multiple communication channels (call and text message, in our case) and we have observed that users organize multidimensional burst in order to minimize the switches between different channels. Finally, we have showed that mobile users allocate their resources – in term of number of interactions – in a different way inside their personal network. Nevertheless, this aspect does not influence the order inside a single burst of activities. This last, counterintuitive, finding is the main motivation to develop a new mobile application that helps users to manage their sociality. The application provides the user with a view of his social life that takes advantage of all its richness and complexity: it organizes his personal network by predicting whom he would like to contact within the current time window, enables user to choose the contact to interact with without worrying about the media to be used, and finally maintains her/his grooming network. These features are achieved by completely on-board computation without any centralized server.

2.4.1 Application functionalities

Media dependency

We are convinced that the mental decision process that determines the sequence of interactions by mobile phone must be driven by social ties. Media

dependencies and switching costs from media to media are operational obstacles hampering natural scheduling. Groo-Me adopts a new HCI model whose attempt is the elimination of these obstacles to bring human interactions back to their natural behaviour. Groo-Me always shows the contacts list ordered by relative importance or strength of the tie. The user selects the person to contact and then selects the channel to use for communication. When the interaction terminates the app autonomously returns at the level of the list. This eliminates the costs of switching and roots the decision process to the sociality.

Grooming network

Groo-Me helps users to maintain their grooming network and becomes an incentive to reinforce strong ties and to explore weak ones. This is possible thanks to an intuitive and manageable user interface fed by on board computation of social relationships. To this purpose Groo-Me records the cumulative amount of calls and text messages per-pairs of users over a sliding window of one month. This feature enables Groo-Me to take into account possible variations of grooming network over a long period. Each personal contact is ranked according to the amount of interactions which is a measure of the intensity of their social relationship. The mere counting of call and text message interactions might rise uneven computations. In fact, text conversations might require a high number of messages sent back and forth between two persons. This could resort to unfairly recording the intensity of a relationship expressed mainly by text messages. To limit this effect Groo-Me clusters all text messages inside a bursts into a single interaction.

The application makes aware the user of her/his current grooming network by graphically reporting her/his grooming contacts ordered by their social strength. In Fig. 2.27 we report a screenshot of a grooming network visualization. Besides, Groo-Me helps users to prevent strong ties from decaying by informing user about the variation of the social importance w.r.t. previous period (in the application prototype we consider the week before). To keep track for users who left the grooming network we provide a "lost" button whose function is to show the list of users exited from the grooming network w.r.t. the week before.

List of favorites

We build a list of favorites which leverages the computed grooming network, user's bursty behavior, and her/his propensity to call a contact in the grooming network approximately at the same time of day. According to the average burst duration, Groo-Me divides the day in one hour long slices. For each time slice Groo-Me assigns to each user in the grooming network her/his likelihood to be called. The likelihood corresponds to the ratio between the number of past interactions occurred in that particular time slice and the total number of

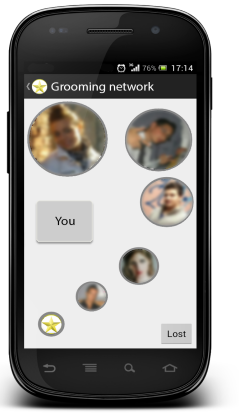


Fig. 2.27: *Groo-Me visualization of a user's grooming network. In this example the size of grooming network is 6 (the smallest bubble has no photo associated to the contact). The bubble size is inversely proportional to the rank. The green circles around the bubble indicate a raise of the users in the grooming network rank. Red and gray circles indicate a decrease or a stabilization of the rank, respectively.*

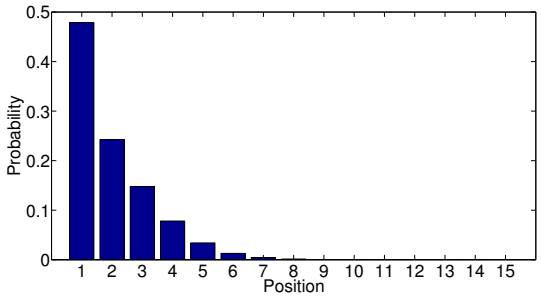


Fig. 2.28: *Histogram of the rank of personal contact in the list of favorites when user called the contact. For each performed call we record the rank of personal contact in the list of favorites if present. Mean 2.01, median 2.00 and standard deviation 1.28*

interactions. Finally, Groo-Me provides a service that dynamically builds a list of favorites by ranking grooming contacts based on aforementioned likelihood.

We evaluated this service by simulating the favorite generation algorithm on the dataset. To this purpose, we preprocessed the dataset to extract the users that were active at least 60 days and we used the first 30 days as training period. As shown in Fig. 2.28, the approach is viable because the call

prediction process actually places the 90% of predicted calls in the highest 4 positions.

2.4.2 Preliminary experiment

We have performed a preliminary experiment by distributing Groo-Me to 12 people, students and Faculty members, and we collected statistics after a period of 2 month together with a survey report asking whether or not the application were able to properly identify the real grooming network. All reports reported that the detected grooming network is actually able to mirror quite accurately the strong ties of their real-life sociality. Moreover, the reported position of the predicted person in the favorite list happens to be on average 1.55, indicating a high prediction level.

2.5 Multidimensional burst generative model

We develop a new generative model which focus on the reproduction of the multidimensional inner structure of each single burst based on the three features d, p and ι that fully describes the multidimensional nature of a burst. The model might be easily interfaced with other generative models which provide values for n (e.g., by drawing from a given distribution as in [65]) and model inter-burst times. We present a generative model for two-dimensional bursts, along with results from a comparison between selected indexes and statistics regarding an excerpt of sample data and a combinatorial analysis of binary sequences.

The generative model of bursts of given length builds on a discrete time Markov Chain (MC), which is instantiated in Fig. 2.29 for the simplest case, $n = 3$. Its regular structure makes it possible to infer the MC appearance for any n. Each MC state represents a (set of) (sub)sequence(s), and may be described by bindings of some integer variables, L (length), P (prevalence), S (switches) and l (last), that take values in $[0, n]$, $[0, n]$, $[0, n-1]$ and $[0, 1]$, respectively. In Fig. 2.29 states are annotated with characterizing bindings. The value of P indicates the number of '1' in (sub)sequences. As for l, it denotes the last event that occurred and allows one to track switches. The one step transition probability matrix of the MC builds on two parameters, pr_0, pr_1, which define the probabilities that the next event in a (sub)sequence coincides with the last one. The values $1 - pr_0$, $1 - pr_1$, thus stand for switch probabilities. We assume that at the beginning of a burst the two possible events occur with same probability.

Some MC states, drawn in bold, correspond to aggregates of (sub)sequences: the higher n, the bigger the impact of aggregation. This way the complexity of the generative model's solution, in terms of number of states, drops from exponential ($O(2^{n+1})$) to polynomial ($O(n^k)$, $2 < k < 3$), so enabling the analysis of bursts of realistic size.

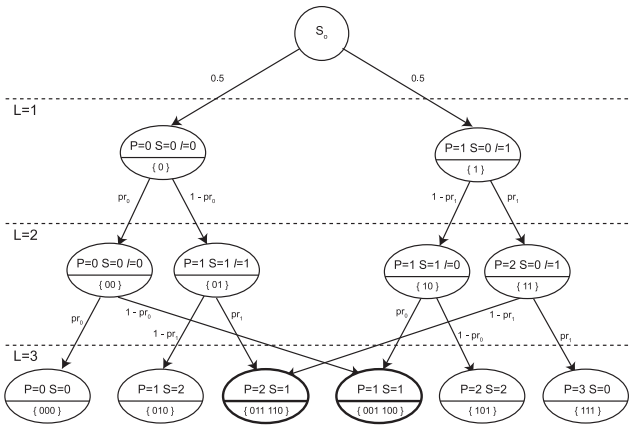


Fig. 2.29: Discrete time Markov chain. *Graphical representation of the discrete time Markov Chain (MC) in case of burst length $n = 3$. MC states are described by four integer variables, L (length), P (prevalence), S (switches) and l (last). L denotes the length of (sub)sequences represented by a state (in the picture values of L correspond to the depth level in the DAG – Direct Acyclic Graph). P and S indicate the number of '1's and the number of switches, respectively. l denotes the last occurred event and allows tracking switches. Each state, but S_0, is annotated with some variables' bindings (top) and the (sub)sequences it represents (bottom). States drawn in bold correspond to aggregates of sequences. The parameters pr_0 and pr_1 represent the probabilities that the next event coincides with the last one. In order to make the MC irreducible, we assume that each final state ($L = 3$) brings the system back to state S_0 with probability 1 (these connections are omitted in the picture).*

However, a direct use of the MC is unfeasible because its size become relevant even for small values of n. The need of identifying a more expressive formalization of the generative model of bi-dimensional bursts as binary sequences led us to consider a timed extension of Petri Nets (PN), known as Generalized Stochastic Petri nets (GSPN) [3, 20]. The GSPN formalism eases the task of modelers by providing compact, parametric and stochastically-reconfigurable representations of even huge Markov processes.

The fact that a GSPN maps to a Markov process, i.e., a Continuous Time MC (CTMC), does not represent a problem. If we consider only the instants at which the state of the system changes, and we number these instances 0, 1, 2, .., then we get a discrete time Markov chain that is called Embedded Markov Chain (EMC), or "jump process", of the CTMC. Letting $q_{i,j}$ be the exponential transition rate from state i to state j of a CTMC, the one-step transition probability matrix of the EMC is defined by setting $p_{i,i} = 0$, $p_{i,j} = \frac{q_{i,j}}{v_i}$ if $i \neq j$, where $v_i = \sum_{k \neq i} q_{i,k}$. The stationary probability distribution $\boldsymbol{\psi}$ of the EMC (which exists and is unique if the EMC is irreducible/positive

recurrent) can be computed from that of the CTMC ($\boldsymbol{\pi}$), and vice-versa: $\psi_i = C\pi_i v_i$, where C is a normalizing constant.

A GSPN generative model parametric in bursts' length was built ensuring that its embedded MC exactly matches the blueprint instantiated in Fig. 2.29 for the case $n = 3$.

2.5.1 GSPN model

The bi-dimensional bursts' generator is depicted in Fig. 2.30, together with an accompanying map legend. The model was edited and analysed using the the GreatSPN package[4], which natively supports GSPN. Let us just overview its blueprint.

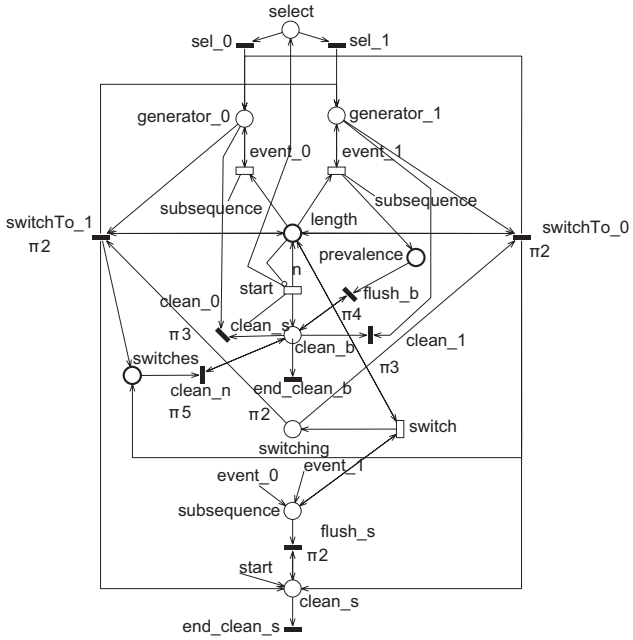
The model implements two event generators that operate in mutual exclusion. Passing from one generator to the other is triggered by a (symmetric) transition which emulates in some sense the switch attitude of humans.

Transition `start` starts the model up by putting n tokens in place `length`, then (once all of them have been consumed, i.e., n events have occurred) brings the model back to its initial state by triggering a sequence of immediate transitions (recognizable from prefixes `clean`, `flush`) clearing the contents of `prevalence` and other significant places. This way the underlying CTMC is ergodic.

Bursts are randomly built by a pair of mutually exclusive timed transitions (`event_0`, `event_1`), one for either type of event (sms/call). The rates of the associated exponential distributions, λ_1, λ_0, may be interpreted as mean frequencies of events of a given type. The ratio $p = \frac{\lambda_1}{\lambda_1+\lambda_0}$ expresses the "average prevalence", and is one of the model's two stochastic parameters. The switching process is explicitly represented, so as to closely adhere to human dynamics: timed transition `switch` indirectly conflicts with both transitions `event_`i, its firing enables one of two (conflicting) immediate transitions (`switchTo_`i) that concretely cause the alternation of the 0 and 1 generators. Places `switches` and `subsequence` hold the number of switches occurred since the beginning of the burst, and the length of the last subsequence, respectively. The contents of place `subsequence` are cleared just after a switch occurrence. Switching is driven by `switch`'s rate, λ_s, the ratio $r = \frac{\lambda_1}{\lambda_s}$ represents the other stochastic parameter of the model.

The parameters pr_i, $i \in \{0, 1\}$, of the EMC associated with the GSPN model are $pr_i = \frac{\lambda_i}{\lambda_i+\lambda_s}$. The metrics of interest are derived from the probability distribution on the subset of states $\{S_i\}$ in which $v_i = 1$ (corresponding to GSPN markings enabling uniquely transition `restart`, that map to the EMC states in which $L = n$). Hence, they can be directly obtained from the GSPN stationary distribution. Referring to the Fig. 2.29, these probabilities may be seen as the probabilities of reaching the final states starting from S_0.

[4] GreatSPN package is available at `http://www.di.unito.it/~greatspn/`



Map legend

Places	
select	non deterministic choice of 1st event
generator_*i*	model operating as *i-generator*
prevalence	burst prevalence
length	burst length
switching	ready-to-switch state
switches	switch counter
subsequence	last sequence length
clean_x	cleaning states
Transitions	
start	model start-up (burst length setting)
sel_*i*	event *i* selected as 1st one
event_*i*	occurence of an event of type *i*
switch	switch attitude
switchTo_*i*	switch to *i-generator*
clean_x, flush_y	clean actions before model restart

Fig. 2.30: **Bi-dimensional burst generative GSPN-model.**

2.5.2 Results

In order to validate the model we performed steady state analysis for a number of configurations. The metrics of interest are the distributions of prevalence and switches (expressed as interleaving degree), which correspond to the distribution of tokens in the homonym places of the GSPN model. Steady-state analysis outcomes are then faced to real data. Besides, we deepen how the

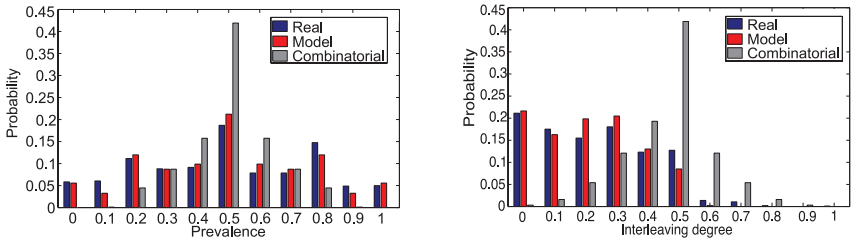


Fig. 2.31: *Comparative analysis in case of burst length 13. (a) Prevalence distributions. (b) Interleaving degree distribution. We use the following parameter values:* $p = 0.5$ *and* $r = 3$.

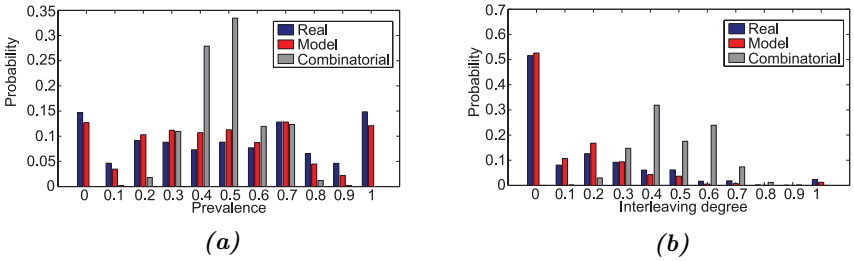


Fig. 2.32: **Comparative analysis on dataset Data-Burst 1.** *(a) Prevalence distribution. (b) Interleaving degree distributions. We use the following parameters values:* $p = 0.5$ *and* $r = 3$.

model performs against a fully random approach relying on sequence combinatorial analysis.

Note that in case of balanced sequences of length n, the formulas for the distribution of prevalence ($P(k, n)$, i.e., the probability to draw k symbols '1') and switches ($P(s, n)$, i.e., the probability to draw s switches) are:

$$P(k,n) = \frac{1}{2^n}\binom{n}{k}\ k : 0 \dots n \ P(s,n) = \frac{1}{2^{n-1}}\binom{n-1}{s}\ s : 0 \dots n-1 \qquad (2.15)$$

The formula for switches is derived from the following consideration. For a sequence of length n, there are at most $n-1$ switches. Then the binomial $\binom{n-1}{s}$ is the number of times in which it is possible to put s switching points among the $n-1$ places.

The search parameter space is given by p ranging from 0.4 to 0.6, r ranging from 3 to 6, which are symmetric intervals around the mean value observed in dataset, and n ranging from 3 to 50. In order to reproduce real prevalence and the interleaving degree distributions, first we selected the best parameters' model using the Kolmogorov distance; then we weighted the obtained per

length distributions according to the burst length distribution observed in the datasets.

In Fig. 2.31 and 2.32 we report the prevalence and interleaving degree distributions of the best fitting model versus the observed and combinatorial ones, both on specific size sequences and aggregated case. There is some evidence that the model fits in sample data much better than combinatorial analysis, which exhibits an unrealistic symmetrical trend with center (approximately) $\frac{n}{2}$. Note that the model is also able to reproduce the peaks due to the one-dimensional and disjoint bursts, which represent the behavior farther from randomness.

The key factor which makes the model closer to the real process w.r.t. the combinatorial approach is its being driven by an explicit switching process able to reproduce the real behavior. Conversely, the random approach is only driven by prevalence parameters, while switching being just an indirect, not tunable, consequence.

2.6 Summary

In this chapter we have analyzed long-range interactions mediated by cellular network by means of two mobile datasets. By using data mining techniques, we have studied different aspects of mobile users behavior such as mobility, burstiness, and sociality. The analysis of the mobility aspects has shown that users exhibit very regular mobility patterns visiting a small set of favorite locations. Moreover, we have analyzed the impact of mobility on phone interactions and we found that a considerable amount of phone calls (around 25%) are performed between persons who are in proximity to each other. Based on these findings, we have proposed a novel LTE-based network architecture placed on the edge network and able to support in-proximity communication channels and local data traffic offload. As a proof of concept, we have considered a newspaper/magazine distribution service. We have performed extensive simulations using real mobility traces extracted from the dataset and we have shown the benefits of the proposed network architecture from both operator and user points of view. We then analyzed the temporal inhomogeneities of mobile users behavior and we have found that users are usually organizing their phone activities in bursts. We have extended previous literature results by considering the multidimensionality of mobile users phone activities. We have developed a theoretical framework suitable to describe the interplay between different communication media (calls and text messages in our case). By applying this framework we have found that mobile users have the tendency to organize their phone activities in order to minimize the switches between different communication channels, rather than follow some order induced by their sociality. Furthermore, we have developed a burst generative model based on GPSN, which is able to correctly reproduce the observed interplay between the communication media. Last, we have analyzed how mobile users manage

their sociality through mobile phones. We found that users interact mostly with a small subset of personal contacts, confirming the Dunbar grooming network theory. Based on this analysis we have developed a new mobile application which helps users to manage their sociality by means of a mobile phone and to keep track of the temporal evolution of their grooming network.

Chapter 3

Short range in urban space

In this chapter we present an Opportunistic Network (ON) deployed on top of Public Transportation System (PTS) to create a fully wireless backbone to support delay tolerant services. We consider a network where nodes are buses, this offers some advantages w.r.t. ONs made up by humans. First, nodes are not subject to battery constraints, second they carry appliances having high computational power and large storage capacity, and third, mobility patterns are highly predictable because buses follow predefined paths and timetables. This last aspect allows routing algorithms based on encounter probabilities, such as [14, 7, 73], to better estimate values of probability since encounters are more predictable.

In the first part of this chapter we discuss the Op-HOP routing algorithm [37] which implements a single copy forwarding mechanism based on encounter probabilities empirically computed by running simulations using real bus lines paths and real timetable. In the second part we discuss the scalability of our protocol against the state of art of routing algorithms specifically design for ONs deployed on PTS. We consider MaxProp [14] and RAPID [7]: two multi-copy algorithms developed for campus PTS named *Dieselnet*. We perform extensive simulation using the PTS of Milan and the results show that Op-HOP is more scalable w.r.t. competitors in term of both delivery time and delivery ratio. In the last part we show the improvement of algorithm performance by introducing a support infrastructure made up by fixed relays placed at bus stops. Each relay is equipped with a radio interface, computational power, and data storage capacity, and it actively cooperates with mobile nodes (buses) to forward packets. We present an algorithm to suitable deploy relays in order to guarantee good network performance and to reduce the costs of the deployment in metropolitan area. Finally, we analyze the network performance of proposed hybrid solution against the pure mobile one by showing that a very limited number of relays can lead to a significant performance improvement in term of delivery delay and delivery ratio.

3.1 Related work

Delay Tolerant Networks (DTNs), as defined by Fall [35], are networks in which links between nodes are not always active. When packet forwarding is not possible, due to intermittent connectivity, the packets are temporary stored in a local buffer until the connectivity is available again. Each node is required to implement routing protocol compliant with the store-carry and forward paradigms. Different number of routing algorithms have been already proposed, as for example [55, 111, 81, 73, 101]. In particular, [73] is a multi-copy forwarding scheme where probabilities are calculated on a per-node base and adjusted with an aging policy. As special cases, in [53, 15, 54, 96, 52] data exchanges taking place with contact opportunity are used to create an ON.

While human-based ONs are still too difficult to be managed due to the unpredictability of human mobility, ONs on top of PTSs have already been deployed in some specific cases. The very first works [87, 25, 26] focused on rural environments in developing regions where buses linked small villages spread over a large territory. In this situation, an opportunistic backbone can be used to provide non time-critical Internet services (e.g., e-mail and non-real time web browsing) to the population.

Later research have been conducted on campus bus networks (e.g., [14, 113, 7]), leading to the design of two most popular routing algorithms for ONs: MaxProp and Rapid. Authors of MaxProp present in [14] a real deployment scenario where, by means of buses, five colleges have been linked with nearby towns and to one another over an area of 150 square miles. MaxProp is proposed a multi-copy routing algorithm based mainly on message priorities. These priorities depend on the path likelihoods to destination nodes on the basis of historical data and other complementary mechanisms. By means of simulation MaxProp have been proven to outperform oracle-based protocols hinged on knowledge of deterministic meetings between peers. This research has been extended in [113], where inter-contact time distribution is analyzed both at bus and line levels. Rapid [7], on the other hand, model the routing system as an allocation problem and try to optimize a specific routing metric such as worst-case delivery delay or the fraction of packets that are delivered within a deadline.

Other contributions targeting bus networks in urban environment are [57, 2, 74, 98], but they are either getting support from a partial network infrastructure or are outperformed by Rapid or Maxprop.

A recent theoretical work about performance analysis for deployment at urban scale is [48]. In this work, authors studied the inter-contact times of the transportations systems of Zurich and Amsterdam, discovering that they follow an exponential distribution. Starting from this result, they predict the performance of a simple epidemic routing using a Markov chain model and demonstrate the feasibility of using a PTS as a DTN backbone in a city.

In two works [8, 115] are presented solutions to enhance the performance of DTN by means of fixed relays, creating an hybrid network made by fixed and mobile nodes.

3.2 Op-HOP routing protocol

Op-HOP (Opportunistic Hopping on Probabilities) is a single-copy probability-driven routing protocol designed to be suitable for a DTN deployed on top of a PTS. Op-HOP builds paths using a probabilistic model based on the number of encounters rather than their durations and opportunistically exploits unplanned encounters to improve delivery performance swerving data on an extemporaneous shorter, or more reliable, path.

3.2.1 Routing on a Probabilistic Graph

The line overlaps of the PTS lead to the construction of a connectivity graph which represents all the possible routes that the messages can take. This problem can be represented as a graph $G = (V, E)$, in which each node $v \in V$ is a bus line and each edge $e \in E$ denotes the existence of at least one connection among buses belonging to the two lines (i.e., the end-point nodes). A path or route r on G represents the sequence of nodes (lines) a message has to traverse in order to be delivered. The weight of an edge is a function of the actual probability of encounter between the two lines.

3.2.2 Encounter Probability Model

Being i and j two bus lines, we denote with $p_{i,j}$ the probability that (any bus of) line i encounters (any bus of) line j. For any bus b traveling on line i, the following Bernoullian random variable $X_b(j)$ is defined:

$$X_b(j) = \begin{cases} 1 \text{ if bus } b \text{ encounters any bus of line } j \\ 0 \text{ otherwise} \end{cases} \tag{3.1}$$

During a day, a bus b makes t_b trips and collects a sample $x_b(j)$ of the random variable $X_b(j)$. At the end of the day the samples collected by all buses of line i are aggregated, and the probability of encounter between line i and j is estimated as:

$$p_{i,j} = \frac{\sum_{b \in i} x_b(j)}{\sum_{b \in i} t_b} \tag{3.2}$$

Considering encounters between different pairs of lines as independent events, the overall probability of a route r, made of d_r hops, is:

$$p_r = \prod_{k=1}^{d_r} p_{k,k+1} \tag{3.3}$$

where k enumerates the nodes along the route r.

Given any pair of source and destination lines, l and m, let $\mathcal{R}_{l,m}$ be the set of all routes $r_{l,m}$ connecting l to m. The best path $r^*_{l,m}$ to deliver a packet from l to m is one in $\mathcal{R}_{l,m}$ maximizing the delivery probability:

$$r^*_{l,m} := \left\{ r^*_{l,m} : p_{r^*_{l,m}} \geq p_{r_{l,m}} \forall r_{l,m} \in \mathcal{R}_{l,m} \right\} \tag{3.4}$$

When applying this probabilistic model on real traces it is possible to create a long path with high probabilities due to a large number of encounters of buses along the path. In such cases, due to the number of forwarding required, the high delivery probability is coupled with a potentially greater delivery delay. Despite delivery probability is the most important metric for us, it does not worth the effort to suffer a significant increase in term of traversed hops in order to raise this probability by a negligible amount. To constraint this phenomenon we decided to keep the probability model as Bernoullian, but to slightly change the probability of encounter estimates by truncating them to the first digit and reducing all probabilities of 1.0 to 0.9.

3.2.3 Construction of the Weighted Graph

An edge between two nodes exists, on the graph G, if the probability of encounter between the two bus lines is greater than zero. To correctly assign weights to edges, we rely on the following chain of equalities:

$$\max_{r \in R} p_r = \max_{r \in R} \log \left(\prod_{k=1}^{d_r} p_{k,k+1} \right) \tag{3.5}$$

$$= \min_{r \in R} \sum_{k=1}^{d_r} \log \left(\frac{1}{p_{k,k+1}} \right) \tag{3.6}$$

Indeed, we define the weight $w_{i,j}$ of the edge connecting node i and j as follows:

$$w_{i,j} = \log \left(\frac{1}{p_{i,j}} \right) \tag{3.7}$$

To find the optimal routes, a shortest path algorithm – such as Dijkstra's – is applied to this probabilistically weighted directed graph $G = (V, E, W)$.

3.2.4 Improving Routing Performance with Opportunistic Contacts

To make the proposed routing algorithm completely opportunistic, forwarding opportunities must be exploited. When a bus b_1 comes in contact with another bus b_2, the routing table is looked up. For every packet in its buffer, b_1 performs forwarding if either: (i) b_2 is the designated next hop, or (ii) b_2 has a shorter distance (or a better delivery probability) to the destination than the designated next hop.

Fig. 3.1: *PTS layout in the city of Milan.*

3.3 Routing protocols scalability comparison

3.3.1 Simulation environment and parameters

Urban Environment

The urban environment we used to evaluate our proposal is the city of Milan (Italy) and its PTS. Milan is a medium size town (typical for many European cities) and its PTS is a complex system extending above and below ground. Due to the underground aquifer and archaeological remains, the subway system is quite underdeveloped. The ground PTS of Milan, spans 69 lines over 49 square miles for a total paths length of 683 miles (13.85 miles for every square mile).

The overall city structure is clearly not Manhattan-like (see Fig. 3.1): crosses between bus lines occur at any time and there is no constant space between intersection points. With this topology, buses run along three kinds of routes: they may span across the city, run around the center, or cover only a peripheral section making a relatively small loop.

While Manhattan-like topologies have been extensively studied in the past, uneven topologies like the one we just described are less addressed. This is why Milan represents an interesting case study: for its uneven topological structure and the frequent contacts between nodes.

Simulation parameters

As simulator we use *URBeS* (Urban Backbone Routing Simulator) which was tested and validated in [37] (for details about the simulator see Appendix A).

In our experiments we considered IEEE 802.11b technology. Available bandwidth is 10 Mpbs and the radio range is 100 meters. Communication takes urban canyons into account; we consider only line-of-sight contacts.

Simulation starts at 4 a.m., with the first bus departing from its head of the line at 5 a.m. and stops at 8 a.m. of the following day, with the las bus going out of service around 6 a.m. All buses departing from their head of the line before 5 a.m. are considered as running during the night and not in the morning.

Data traffic generation is performed continuously during working hours: from 8 a.m. to 8 p.m. During simulations each operating bus generates a stream of 64 KB unicast packets at a constant rate; each packet will be delivered to a distinct, randomly chosen, bus line.

Data packet generation rate will vary from 20 to 80 packets per hour from every bus. The total number of packets coming into play will thus be between $138,364$ (20 packets per hour) and $553,396$ (80 packets per hour). When a packet is generated it is placed in the node local buffer until forwarding becomes possible; an isolated bus will keep accumulating packets while no contacts are experienced. When an encounter happens all the packets are checked for forwarding and transmitted in accordance with the adopted routing policy if the encounter matches a positive opportunity. Bandwidth is accounted using a token bucket mechanism while buffer space availability is simply checked before transmission. In case of contention a first-in-first-out policy is applied. The size of the application buffer on board at every bus has been set to 4 GB.

When a bus reaches the end of the line it may or may not queue up and wait for another scheduled departure. If the bus stays in line it will hold all its data and will keep generating packets while waiting. If, on the other hand, the bus leaves service all content will be pushed to the first bus waiting in line. If there is no bus (because there are no more scheduled departures) all the stored packets are dropped and packets will be considered lost.

3.3.2 Simulation results

Resources Usage

In our study we deal with two types of resources: buffer space on nodes, and bandwidth availability.

For peak traffic, global buffer space allocation is reported in Fig. 3.2. As it can be noted, with high level of traffic replication take a significant toll on resources: to manage a total of $553,396$ during a whole day we must manage up to 8 millions replicas (almost 15 times as much) over around 600 buses running at 8 p.m. This is due to the extremely high number of buses running in a real urban environment and the frequent contacts in a complex topology. Please, while reading the figures, consider the difference in scale: the average allocation at the end of simulation must be multiplied by the number of buses

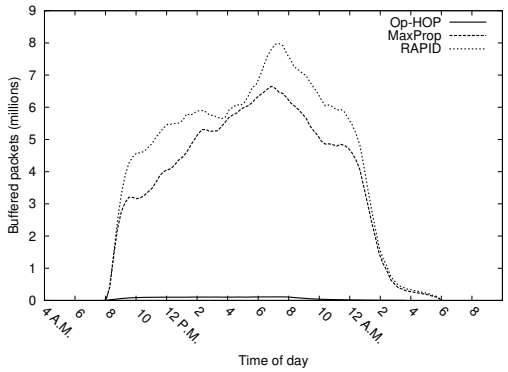


Fig. 3.2: *Total number of buffered packets at 80 packets per hour from every bus.*

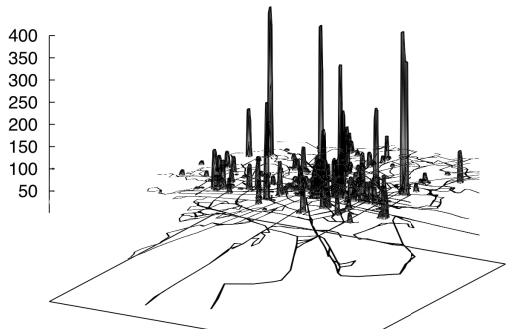


Fig. 3.3: *Count of missed forwarding opportunities by location with 20 packets per hour from every bus.*

(which is going to 0) while Fig. 3.2 counts packets in millions; this is why one figure is diverging while the other one is not.

Op-HOP, as a single copy algorithm, has a negligible impact on buffer usage and its profile can be barely seen in the graphs.

Nevertheless, local buffer allocation is not a problem even for multi-copy: on peak usage we account for an average of 1.6 GB of used space. The real problem when using multi-copy is related to the required bandwidth, which translates to missed forwarding opportunities: nodes with overloaded buffers are not be able to spool all packets during a single contact. Fig. 3.3 reports a count, by physical location, of contacts which have been too short to allow for a complete spool. Note, for every missed opportunity many packets are affected. For sake of readability, in this case we used the lowest traffic profile. In average, 13% of packets missing a forwarding opportunity will not get delivered. Waiting for another good contact may require a full run, adding

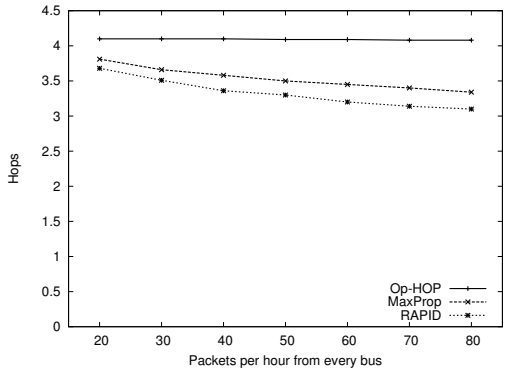


Fig. 3.4: *Average number of traversed hops.*

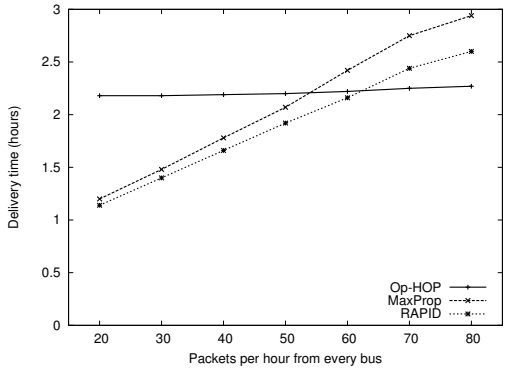


Fig. 3.5: *Average delivery delay.*

delay, and packets with too much delay will get discarded at the end of the day.

As a result, keeping buffer usage at a minimum seems to be a winning strategy for a urban DTN. This is vouching for the use of a single-copy approach.

Performance Evaluation

The performance metrics we are interested in measuring are delivery delay and delivery rate. In a DTN, the number of traversed hops greatly affects the delay. As we can see in Fig. 3.4, the average number of traversed hops with increasing traffic decreases for multi-copy routing and is essentially stable for the single-copy one. This seems to be a good result, but actually it is not: the number of hops is decreasing because the delivery delay is increasing a lot (more than 2 hours, see Fig. 3.5). Packets getting delayed too much are

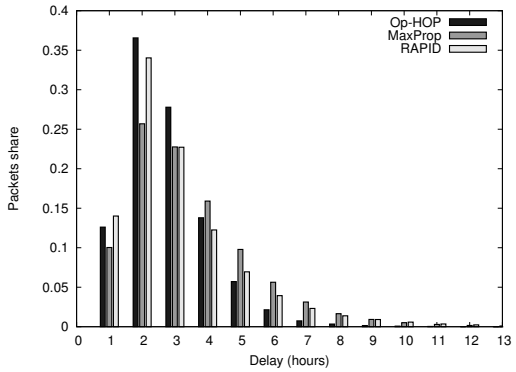


Fig. 3.6: *Distribution of delivery delay at 80 packets per hour from every bus.*

discarded at the end of the day and does not account for this statistic. On the other hand, our single-copy approach keeps a steady behavior both for the number of hops and the delivery delay. Once again, the answer sits on the buffer usage: packets which cannot get forwarded on the right line are required to wait for another run or to take a longer (non optimal) route to the destination. The delay distribution histogram (Fig. 3.6) offers an interesting proof for this: MaxProp, which is the algorithm with the greatest number of replicas has the greater share of packets getting delivered in 4 hours or more. Op-HOP, on the other hand, increases delivery delay by a limited amount and exposes a good scalability behavior.

Delivery ratio gets also affected: in Fig. 3.7 we can observe that starting from 60 packets per hour from every bus the single copy approach outperforms the multi-copy ones.

It is worth to be noted that multi-copy approaches do not reach 100% delivery ratio anyway (they top around 97%). This is caused by packets left on buses at night not getting delivered due the extremely disconnected network; of course, having multiple copies of the same packet limits this phenomenon to traffic generated during late evening because all copies have to miss their destination. On the other hand, single-copy routing is more prone to undelivered packets due to critical paths with low probability but will not suffer from scalability issues on high traffic.

Instead, looking at low traffic simulations, single-copy approaches seem to have lower delivery ratio. This is due to critical paths with relatively low probability,

All performance metrics seems to indicate that multi-copy algorithms are not scaling as good as single-copy one with increasing traffic. In particular, our protocol, Op-HOP, seems to react very well to the complexity of the large city and the offered load.

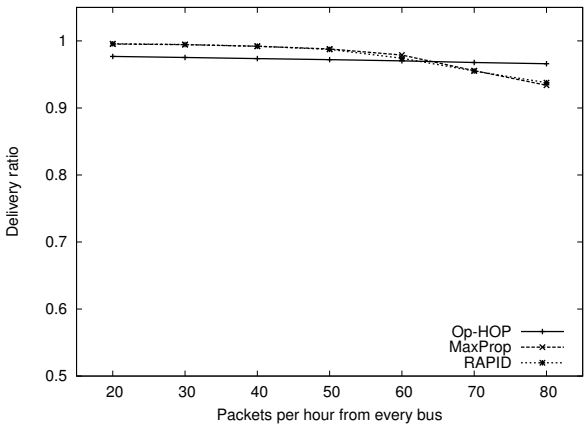


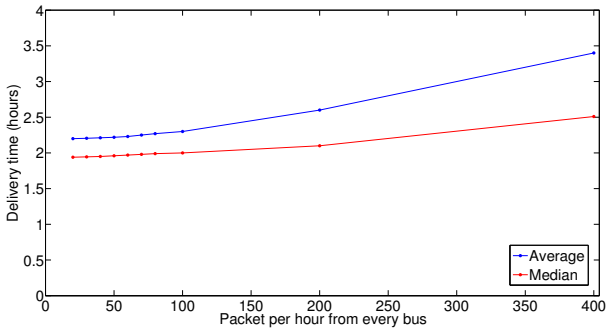
Fig. 3.7: *Observed delivery rate.*

3.3.3 Op-HOP performance limits

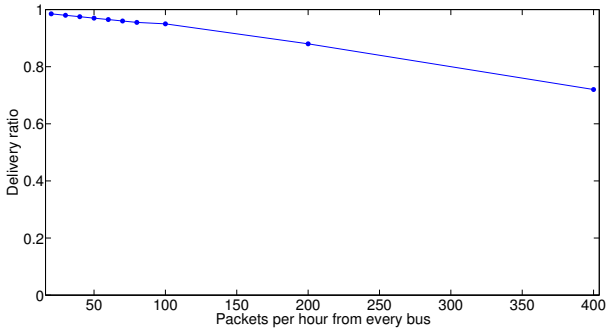
We have just seen that Op-HOP outperforms multi-copy routing algorithms. Here we want to stress our protocol to find its limits in terms of scalability. We perform a new set of simulations increasing the packets generation rate, from 100 to 400 packets per hour generated by every active bus. The simulations results show the decrease of performances of Op-HOP. In Fig. 3.8a we can observe that delivery time increases reaching a mean value of 3.5 hours and a median value of 2.5 hours in case of maximum packets generation rate. Moreover, in Fig. 3.8b we observe a drop in delivery ratio performance, going to around 70% with highest level of packets generation. These two results clearly show performance issues with very high level of data traffic. We notice that performance deteriorates especially during the late hours (after 8 p.m.) due to the decrease of running buses. We found that this problem is more noticeable in case of bus lines having only one neighbor line (neighbors distribution is reported in Fig. 3.9) with limited cross points, because if a forwarding opportunity is missed packets need to wait the completion of an entire trip to have a new chance to be forwarded.

3.4 Supporting infrastructure

In the previous section we have shown the performance issue of Op-HOP routing algorithms at very high level of packets generation rate. We overcome this problem by introducing a fixed road side infrastructure made up by fixed relays equipped with WiFi interface and data storage. In this section we provide a comprehensive definition of our proposed infrastructure and we develop a suitable algorithm to deploy and minimize the required relays. By means of



(a)



(b)

Fig. 3.8: *Mean and median of delivery time (a). Delivery ratio (b).*

extensive simulations we will show that a very limited number of road-side units can lead to a significant performance improvement in term of delivery delay and delivery ratio.

3.4.1 Infrastructure Deployment

Here we are going to introduce the road-side network infrastructure and describe our algorithm to select infrastructure nodes position.

Road-Side Exchange Points

Our road-side infrastructure will be made of apparatus we baptized Road-Side eXchange Points (RSXPs). RSXPs are wireless-equipped appliances located at bus stops and working as fixed disconnected relays. Operating in a urban area, RSXPs are supposed to be permanently attached to power supply and can adopt complex hardware to manage substantial data flows and store a large amount of data in the internal memory.

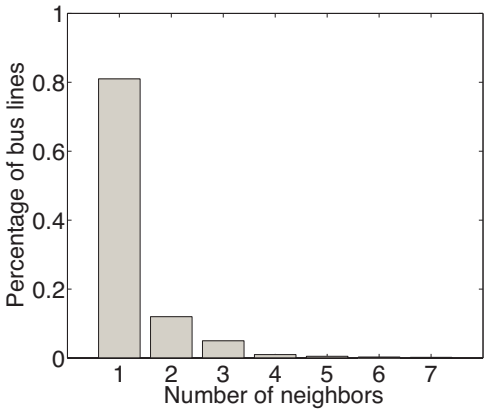


Fig. 3.9: *Histogram of the number of bus line neighbors per each bus line.*

An RSXP will build an internal database of lines passing nearby. When a bus comes into transmission range with an RSXP the following will happen:

1. The RSXP will declare to the bus all known lines.
2. The bus may decide, based on its forwarding policy, to send to the RSXP packets whose next hop is known.
3. The RSXP will send to the bus all packets for which it is a valid next hop.

As we can see, the routing decision sits only on the bus, and the RSXP will act as an external cache; changing routing policy will not require any upgrade or management on the road-side infrastructure. Moreover, RSXPs are not known to the buses: they are exploited when encountered in the same fashion as a contact opportunity with another bus. This way, a faulty RSXP will not jeopardize data forwarding on the DTN.

Reduction Algorithm Overview

The reduction algorithm we propose is composed of three phases.

1. Bus stops covering the same radio area are merged.
2. Relevant bus stops are identified from lines encounter probability.
3. Redundancy is eliminated by solving a Set-Covering optimization problem.

In the following Sections we describe in detail the three phases of the algorithm.

Sites Selection

In order to select deployment sites for RSXPs we start from bus stop positions. At bus stops we have scheduled pulls up where a longer intra-contact

time is experienced and a significant data transfer can take place. Moreover, intersecting lines share the same bus stop, making the place a good location for a packet waiting for a ride. Of course, placing an RSXP at every bus stop will create a huge overlap and, as a consequence, a huge waste of resources.

In order to reduce the number of RSXP we implement the following greedy algorithm.

1. An RSXP is created at each bus stop and is tagged with all bus lines stopping there.
2. For every RSXP its neighborhood is analyzed and tagged lines are merged. Two RSXPs are neighbors if and only if they are in mutual transmission range and in line of sight.
3. RSXPs are sorted in descending order, based on the number of the distinct lines the RSXP has in sight. The list is then scanned starting from the first element; the algorithm removes all other RSXPs which are neighborhood of current RSXP.
4. All RSXPs tagged with only one line are discarded because they will not offer any real contact opportunity.

This algorithm guarantees that resulting RSXPs have all declared lines in-sight, because if two stops of different lines are in-sight, also buses belonging to those lines have both stops in-sight.

When this approach is applied to the city of Milan, the total number of RSXPs is reduced from 3741 to 561 with a reduction of 85% for the number of sites.

Required RSXPs

The number of RSXP can be further reduced based on the lines encounter probability. We want to use the road-side infrastructure to forward data only between those pairs of lines which have a low encounter probability. The encounter probability between pairs of lines is calculated as in Section 3.2, where a Binomial model based on history of buses encounters is used. This way, we will boost probability only for those hops who are less likely to happen.

We define an RSXP to be required if and only if there is at least a pair of lines which can be reached by the RSXP whose encounter probability is below a certain threshold, τ. Only pairs of lines having a non-zero encounter probability are considered, to avoid the case when two lines never encounter. More formally, given L the set of all lines in the system, and L_x the set of lines reachable from RSXP x ($L_x \subseteq L$) we can define the set U as the set of all pairs of line whose encounter probability is lower than τ.

$$U = \{\{u, v\} \mid p\,(u, v) \leq \tau, u \in L, v \in L, \tau \in (0, 1]\} \tag{3.8}$$

RSXP x is required if a pair $\{u, v\} \in U$ such that u and $v \in L_x$ exist.

By applying this criterion we can tune our system to the best value for τ with respect to the observed performances.

For the city of Milan, the number of required RSXPs depending on τ is reported in Table 3.1, second column.

Further Reduction

One last level of refinement can be reached: it is possible that the same pair of lines is covered by more than one RSXP. This means that potentially we might deploy more RSXPs than the ones needed to cover the set U.

This last reduction has been obtained by solving a set covering problem on U. Given R as the set of required RSXP, the incidence matrix A can be defined as:

$$A = [a_{i,j}] \qquad i \in U, j \in R$$
$$a_{i,j} = \begin{cases} 1 & \text{if RSXP } j \text{ can see both buses of pair } i \in U \\ 0 & \text{otherwise} \end{cases} \tag{3.9}$$

Then we must find

$$min \sum_{j \in R} X_j \tag{3.10}$$

with

$$\sum_{j \in R} a_{i,j} X_j \geq 1 \qquad \forall i \in U, X_j \in \{0,1\} \tag{3.11}$$

In the formulas, X_j represents the binary variable associated to RSXP j.

The third column of Table 3.1 reports the number of RSXPs for the city of Milan when applying the complete algorithm.

τ	Required RSXPs	Deployed RSXPs
0.1	30	17
0.2	78	47
0.3	128	79
0.4	168	101
0.5	198	110

Table 3.1: *Number of RSXPs for several values of τ.*

3.4.2 Performance analysis

Simulation parameter

Data traffic generation is performed continuously during working hours: from 8 a.m. to 8 p.m. During simulations, unicast traffic is generated from <*one bus of a bus line*> to <*any bus of the destination bus line*>, following a rationale

discussed in [113]. Each operating bus generates 64 kB packets at a constant rate; each packet will be delivered to a distinct, randomly chosen, bus line. Data packet generation will vary from 20 to 400 packets per hour from every bus. The total number of packets coming into play is thus between $138,364$ (20 packets per hour) and $2,766,965$ (400 packets per hour).

When a packet is generated it is placed in the node local buffer until forwarding becomes possible; an isolated bus will keep accumulating packets while no contacts are experienced. When an encounter happens all the packets are checked for forwarding. Forwarding is, of course, subject to bandwidth and buffer limitations. Bandwidth is accounted using a token bucket mechanism while buffer space availability is simply checked before transmission. In case of contention a first-in-first-out policy is applied. The size of the application buffer on board at every bus has been set to 4 GB. Packets are then forwarded in accordance with the adopted routing policy.

When a bus reaches the end of the line it may or may not queue up and wait for another scheduled departure. If the bus keeps a place in line it will hold all its data and will keep generating packets while waiting. If, on the other hand, the bus leaves service all content will be pushed to the first bus waiting in line. If there is no bus in line (because there are no more scheduled departures) all stored packets are dropped and will be considered lost.

When a packet is queued for forwarding, the adopted routing policy is used to identify the next hop. Packet forwarding can take place under three different conditions: (*i*) the bus which is carrying the packet encounters a bus belonging to the next-hop line, (*ii*) the carrying bus encounter an RSXP which has in its database the next-hop line for the packet, and (*iii*) if the packet is queued on an RSXP and a bus belonging to the next-hop line shows up. We adopt Op-HOP as routing policy in order to better asses the actual gain provided from the infrastructure.

Simulation results

In this section we are going to discuss the outcome of our simulation experiments. We will present Op-HOP performance improvements with respect to various values of τ, including 0 and 1. When τ is 0, pure ad hoc mode will be adopted: no RSXP will be deemed as required and only bus-to-bus forwarding will take place. This value will provide a baseline for performance comparison. On the opposite, if τ is 1, all RSXPs will be required and all forwarding operations will be performed through the infrastructure: bus-to-bus forwarding is forbidden. Experiments with this value of τ will provide a performance upper bound.

Data Delivery

Unsurprisingly, exploiting the network infrastructure is reducing the average delivery delay, as reported in Fig. 3.10a. From this figure, we can observe a

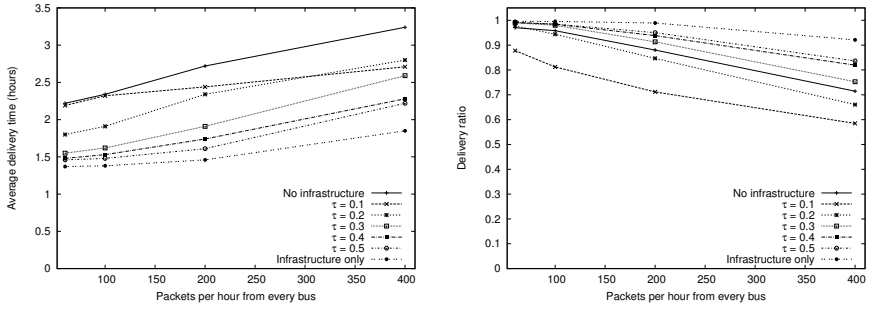


Fig. 3.10: *Performances at increasing offered load with multiple value of τ: average delay (a) and average delivery ratio (b).*

significative improvement, up to one hour (i.e., 30%), even with low values of τ. Interestingly enough, increasing the offered load seems to make the situation better thanks to packets sitting on RSXPs and leaving in bursts for the next hop. This behavior can also be partially explained observing the average delivery ratio (Fig. 3.10b). By combining Fig. 3.10a and Fig. 3.10b we can see that packets following long paths gets discarded more frequently. Discarding packets running long distances reduces the average delivery time for the ones who are able to reach the destination.

Moreover, lower values of τ seem to make delivery ratio worst while for $\tau \geq 0.3$ the situation improves. This behavior indicates that, depending on the PTS, there is a lower bound for the number of RSXPs to deploy. Below this bound, the infrastructure is not able to manage all the offered load. The explanation is that overloaded RSXPs will not be able to spool out all queued packets during a single bus contact, increasing significantly the buffering time and leading to considerable packet drops at the end of the day.

Furthermore, we detected an uneven usage for the RSXP buffers. As an example, in the worst case, with $\tau = 0.1$ (17 RSXPs) at peak traffic (6:30 p.m.) we have 14% of RSXPs whose buffer is empty and 23% of them who are not able to spool all queues in a single bus encounter. On the other hand, the most uniform situation can be observed with $\tau = 0.5$. As a matter of fact, for the PTS of Milan, a good tradeoff value for τ seems to be 0.4. In all the examined cases the system seems to perform better than the infrastructure-less version and setting τ to 0.5 does not give a significative improvement over 0.4.

Number of Hops

The average number of hops for delivered packets is reported in Fig. 3.11a. As we can see in the picture, for low values of τ the use of an infrastructure makes the paths longer. This is confirming the performance degradation outlined in

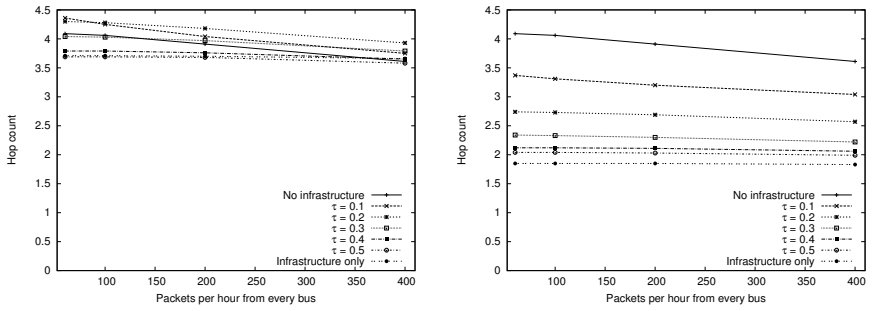


Fig. 3.11: *Statistics on delivered packets: average number of hops (a) and average number of involved lines (b).*

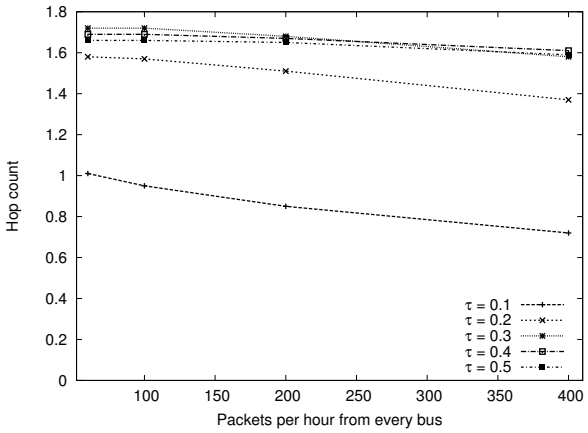


Fig. 3.12: *Average humber of hops between buses and RSXPs for delivered packets.*

Sec. 3.4.2. Greater values of τ improve the situation but do not outperform the infrastructure-less version in a sensible way when the traffic is very high.

Obviously, if we consider only the lines traversed by a packet, the paths are actually becoming much shorter, as depicted by Fig. 3.11b. This is due to the fact that short paths traversing the infrastructure benefit from contacts opportunity happening for sure (sooner or later). These short paths will no longer get penalized in favor of longer, and more probable, ones. Nevertheless, here we can observe that the reduction of the number of involved lines is not proportional to τ or the number of RSXPs. Once again, the answer can be found in an uneven distribution of the network load: full buffers at RSXPs will prevent some buses to use the shortest, planned, route and will make them wait for a bus-to-bus forwarding opportunity.

Another interesting consideration comes from Fig. 3.12, where the average number of hops not taking place between buses is reported. While it is obvious that increasing τ we increase the number of RSXP and, in turn, the degree of involvement of the infrastructure; it is less obvious that there is an upper bound for the number of hops provided by the infrastructure regardless of the number of RSXPs. In our case, the average infrastructure involvement is not greater than 1.75 hops for any value of τ; unless, of course, we forbid data forwarding between buses.

By comparing Fig. 3.11a and Fig. 3.12 we can observe that, for $\tau \geq 0.3$, 50% of the hops are from bus to RSXP (or vice versa). Looking to Fig. 3.11b, for the same values of τ, the average number of lines traversed by a packet is 2. Putting these information together we can state that, after a certain density of RSXPs (i.e., a certain threshold for τ) the average path converges toward a sequence like "*source line* $\rightarrow$ *RSXP* $\rightarrow$ *destination line*", and deploying RSXPs above such density is a potential waste of resources. This is the explanation for the upper bound in Fig. 3.12.

From the figures above, in the case of Milan, we can confirm that setting τ to 0.4 gives a good tradeoff between performances and infrastructure deployment.

3.5 Summary

In this chapter we have considered medium- and short-range interactions mediated by Wi-Fi radio technology. We have exploited this kind of interactions to create an Opportunistic Network (ON) deployed on top of Public Transportation System (PTS) suitable to support delay tolerant applications and services. By means of extensive simulations on the actual PTS of the city of Milan we have shown that the single copy routing algorithm Op-HOP scales at urban level avoiding the overhead of multiple replicas which affects MaxProp and RAPID. Then, we have addressed the allocation problem for a network infrastructure to augment the overall ON performance. The proposed solution consists in the creation of a road-side network infrastructure by means of disconnected relays we call Road-Side eXchange Points (RSXP). Using a suitable algorithm, we identify a proper subset of bus stops candidate to take part in fixed infrastructure. By means of extensive simulations we have shown the impact of RSXPs density on the DTN performances. Results indicate that delivery delay will benefit from the presence of the infrastructure but, in order to improve the delivery ratio, a minimum density is required. The number of traversed hops indicates that the infrastructure involvement in the forwarding process is upper bounded and deployment of RSXPs above a certain density is a potential waste of resources. Using these results, we have demonstrated that a lightweight, and non invasive, road-side infrastructure is a feasible way to deploy a DTN over a PTS to provide mobile applications and services based on location and context on a urban scale.

Chapter 4

Micro range in urban space and the Internet of Things

Mobility in urban environment allows users to interact with objects present in a metropolitan area, e.g., museums, monuments, shops, and other points of interest.

The social graphs describing today's social networks are quickly densifying. This is progressively draining the value of online social networks due to the widespread presence of no worth ties. Moreover, it is also raising concerns, about spam control and preservation of user's image against infringements deriving from posts issued by untrusted 'friends'. Despite the frequent encouragements to clean up personal links, the growth mechanism of today social networks is unable to guarantee a trusted space where persons can reinforce their friendship and enlarge their social capital.

THINPLE is a novel mobile computing app, aiming at building a protected and trusted online social space as the mirror of the real life sociality manifested by physical encounters. By viewing the phone as an extension of the user, THINPLE exploits each physical contact, through the short radio range of Near Field Communication (NFC). NFC is a next-generation short-range wireless communication technology enabling the exchange of data between devices brought within a 4 centimetres distance. Thanks to this very short communication range, it allows us to capture the explicit willingness and trustworthiness to mirror such a contact event into the personal online sociality. Besides, this feature is enforced by Android system policies which disables the NFC radio interface when the mobile phone is in standby. The THINPLE's approach, unlike current social networks, perfectly recreates real life friendship creation phase as a synchronous event involving both parties. At contact time, people agree about friendship, thus avoiding individuals getting bored about spam of friendship requests. In THINPLE, the growth of the social space of a person is only fuelled by contacts in the physical world.

This basic idea has another great and unexplored potential: through physical contacts. Inanimate objects of the real world can be added into the online

sociality of individuals, thus turning into practice the concept of Internet of things [6] and people. By equipping objects with NFC tags, a contact with an object explicitly adds it to our space of sociality. As a consequence, restaurants and shops, monuments and points of interest, as well as home appliances or personal belongings can enter the social graph of individuals and communicate through trusted channels to provide a variety of services: from fidelity programs and advertisement to remote control and product traceability. Moreover, THINPLE makes possible the creation of emerging communities around inanimate objects. These communities are based on a shared real interest which makes itself visible through the creation of a link on the online social network.

THINPLE provides a unifying framework where people and things are seamlessly combined into a common graph. For this reason, the name THINPLE is obtained from crasis of words things and people. THINPLE is inherently a mobile application complemented by cloud services. It aims at the following:

- create and manage the social capital of individuals as composed of trusted people and loved objects;
- ensure a protected space where social interactions are conducted and the personal social capital can be expanded;
- enable service providers to deploy their applications on the top of THINPLE social network.

4.1 Related work

The scientific community has been attracted to study the human sociality in order to better understand its leading mechanisms. The knowledge about these mechanisms can be exploited to offer services leveraging the social information, and to guide the design of new network paradigms as opportunistic networks, which exploit unplanned encounters to route data among the devices [89, 90, 91]. In the last years many efforts have been done to identify the real human encounters. They have been inferred using devices (equipped with Bluetooth, GPS, and Wi-Fi) proximity [29, 38, 39] or self reported surveys [51, 30]. However these experiments only capture the people proximity, but lack in capturing the explicit willingness of a meeting. Smartphones equipped with NFC allow us to overcome this limitation. Up to now NFC has been exploited to exchange personal information or content sharing between people [27], and between people and object, as for payments. To the best of our knowledge, this is the first social network based on real encounters, and including both people and objects.

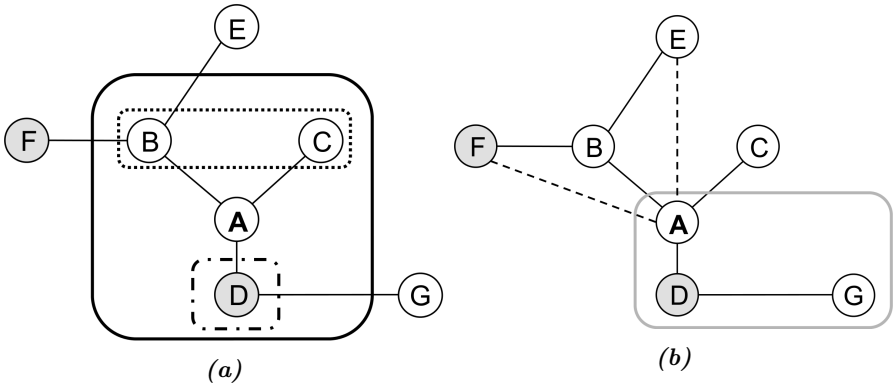


Fig. 4.1: *An example of THINPLE graph. a) White nodes represent people and grey nodes represent things. The social capital of the node A (solid box) is the union of their human (dotted box) and environmental (dashed box) capitals. Solid lines represent links happened as a result of real encounters. b) Dashed links represent two hops potential friendships mediated by the person B, while the node (thing) D is the mediator for the community (gray box) grouping nodes A and G.*

4.2 The online sociality of THINPLE

THINPLE adopts a novel creation and growth policy and applies it to the model of online sociality that is adopted by most of today's social networks. The THINPLE's aim is expressing the social potential of people and things by leveraging contacts in the real world to make explicit friendship requests or interest towards a thing. This contact-based growth policy allows us to create the scenario of Fig. 4.1a, where each person, let's say A, B, C, E and G (white nodes), is represented by a node in a graph with links to neighbor nodes associated to encountered friends. In this view the node neighborhood represents the real human capital of a person and mirrors his/her sociality. Inanimate objects, let's say D and F (grey nodes), surrounding a person and catching his/her interest, can also enter the graph. They become the environmental capital of a person. As a consequence each person is characterized by a *social* capital (solid line in Fig. 4.1a) which is the union of *human* (dotted box in Fig. 4.1a) and *environmental* capitals (dashed box in Fig. 4.1a).

As we can see in Fig. 4.1b, a person can expand the borders of his/her social capital by exploiting the his/her two-hops view as follows. As mentioned before the persons' one-hop neighborhood (the social capital) is made up by things (the environmental capital) and people (the human capital). The two-hop viewing allows a user to look at the social capital of his/her human capital only; in a nutshell you can see the friends and the objects of all your friends, but you cannot see people who tapped on the same object. This choice is motivated by observing that a two-hops view of people who tapped on an

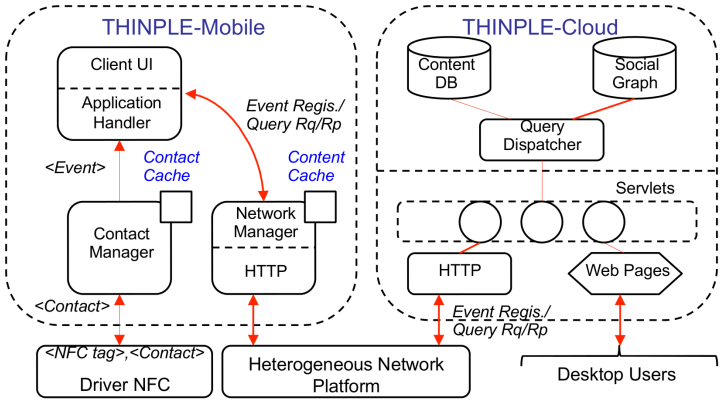


Fig. 4.2: *The architecture of THINPLE-mobile and THINPLE-cloud.*

object (for instance, a monument) is mostly useless (and often unaffordable). For instance, A is aware of B and C neighbors (E and F), but s/he can't see D neighbors (G). This feature allows the creation of weak connections representing potential new friendships. For instance, a node, say A, may use a weak connection (dashed link), *i.e.,* link towards a 2-hop node, to E to plan an encounter, that still remains the only mechanism to let E enter the A's human capital (turning the dashed into solid link). As regards environmental capital, the two-hop viewing makes easier the growing of personal environmental capital. For example, should F be a monument, by viewing F in the B's space, A may be encouraged to visit it, tap on the NFC tag of F thus letting F to enter his/her environmental capital.

In our design a thing's node has a functionality that a person's node has not: it can act as a mediator to form a community of people sharing the same interest about the thing (solid grey box in Fig. 4.1b). Many of those could be willing to share comments. To appease this wish, a person can join the community that has been opened by the object. All communications inside a community rely on the mediation of the relevant object and are not performed through dashed links, as happening between persons.

In THINPLE the Internet of people and the Internet of things are merged and linked by relationships that model real life attitude to manage and expand personal human and environmental capital.

4.3 The functional architecture of THINPLE

In this section we present the functional architecture of THINPLE platform. The architecture has been designed in order to easily allow the service providers to deploy their application on the top of THINPLE platform. To accomplish this task we build the THINPLE architecture maximizing the

modularity and reusability of the different components. Another design guideline is the efficient management of the social graph, representing the social network, and the contents associated to nodes and their relationships.

At the moment, THINPLE relies on a centralized social graph. In fact, our first aim is the construction of a new sociality leaving privacy concerns and disintermediation issues to a future phase. THINPLE has two functional components: THINPLE-mobile and THINPLE-cloud (Fig. 4.2).

4.3.1 THINPLE-Mobile

The THINPLE-mobile component includes the following three modules:

- **Contact Manager** - this component receives the NFC contact descriptor and generates the associated event that is described by the tuple *<nodeID, timestamp, [public key], [encounter ticket]>*. While the timestamp is generated locally, all the other (mandatory and optional - in brakets) data are exchanged at contact time. Both persons and things are identified by a unique nodeID, obtained in a similar way of IMSI number of mobile phones and is composed of three parts: *type, country code, provider code, tag number*. The first is used to distinguish people and things; the second is unused at the moment and accounts for future possible organization by region; the third is set to 001 for THINPLE and could be useful when different service providers will be using the same THINPLE platform. The last is a progressive integer. This nodeID uniquely identifies it inside the social graph on the THINPLE-cloud.
  The Contact Manager has to manage two types of contact: among people, and between people and things. In the first contact type, the mobile phone is reader and sender of contact descriptor at the same time. In order to successfully complete the exchange, both devices have to simultaneously send their contact descriptors. Unfortunately, the end-devices of a NFC channel are rarely synchronized. Due to the current Android NFC policy, in such an asynchronous exchange the first sending device happens to block the sending of the other one. To prevent this event, the Contact Manager receiving the first NFC contact descriptor starts a timer. Should the timer expire without closing the exchange, the Contact Manger transparently opens a Bluetooth channel to complete it. At this point, both peers are autonomously able to update their human capital on THINPLE-Cloud. In the second contact type, the mobile phone is acting as NFC-reader and it has the responsibility to update the users's environmental capital. Each contact event is locally cached to avoid multi event occurrences due to multiple tapping;

- **Network Manager** - It realizes the Internet (cellular and WiFi networks) connectivity to THINPLE-Cloud. The network channel (WiFi or 3G) is

automatically selected by Android operating system. The Network Manager negotiates with THINPLE-Cloud which contents to download. The Network Manager sends to server the current channel condition, using the system API, and the THINPLE-Cloud selects the contents to send to mobile, according to its policy. For instance, if we are in presence of a WiFi network (a very good network condition with respect to 3G network), the server decides to send all contents to mobile (text, images, video, and more). If we are in presence of 3G network, the server decides to send to mobile the lighter part of contents (text and small images), thus improving the responsiveness of the application. The content transfer is operated through HTTP/HTTPS sessions. A local cache maintains the most recent downloaded contents to avoid unnecessary network access may cause waste of energy. In absence of radio connectivity, pending requests to THINPLE-Cloud are temporarily queued locally and and they will be sent to THINPLE-Cloud as soon as the connectivity became available.

- **Client Application** - the component has two modules. The Application Handler provides a unifying API interface towards the module Client GUI to ensure transparency with respect to the underlying communication channels. The Application GUI implements the user interface of THINPLE. Moreover it is responsible to orchestrate the below components.

4.3.2 THINPLE-Cloud

On the server side, the THINPLE-cloud component includes:

- **Server applets** - A set of JAVA Servlets manages all interactions with the client side, through HTTP/HTTPS sessions, and with desktop users. Each Servlet may generate requests to the Query Dispatcher module whenever data have to be retrieved. The interface between the two modules has been designed to ensure layer independence.
- **Social Graph** - This component is responsible to manage the social network using a Graph Database instead a Relational Database in order to efficiently navigate into the social graph.
- **Content DB** - This module manages a Relational Database containing all contents regarding the node (name, photos, ...) and the relationships (date of meeting, shared content, meeting place, ...).
- **Query Dispatcher** - According to the type of the query this module routes requests to the Social Graph (for instance, when it is required to navigate the graph) or to the Content DB (for instance, to retrieve the contents associated to a given object), and forwards replies to the proper Servlet. The contents associated to graph nodes and social graph are kept separated to satisfy scalability and performance issues.

4.4 Enabling new mobile applications

The aggregation of people and things into a unifying social network is the richness of THINPLE and enables or improves a variety of challenging mobile applications and services beyond the traditional services of online social network. It is in fact very simple to imagine a variety of application scenarios centered around things, as briefly shown in the sequel.

- Example 1 - Let's imagine that F is actually the bike of B. This enables F to be present into the B's space with general information and photos, but also to deliver workout data for comprehensive off line analysis (see, for instance, GarminConnect).
- Example 2 - Let's now assume that the object D is a pub, or an art gallery or a temporary shop. Each person who visited it at least once has a link to it and all friends have a two-hops view of it. The owner of D can advertise a forthcoming event that can be viewed two-hops apart. To open a window onto the future, THINPLE powers the *"I'll go"* button that enables interested people to state the willingness to attend the event. This would be useful not only to the owner of D, but also to favor people aggregation.

In the following subsections we provide some more details about two novel mobile applications we are working on and that are currently in a prototype stage.

4.4.1 Innovating the Grand Tour approach to travels

The approach to cultural travels has been unchanged for almost three centuries now. Both modern and Grand Tour travelers mine their voyage information from books and guides. Today, several institutions rely upon the web to promote their cultural heritage; the web, however, is a great information source mainly while preparing the trip. On the other hand, the attempts to port guides on mobile devices went almost flat as unable to add value in terms of both usability and functionalities. There is nothing that can be done to complement traditional travel supports and to improve the travel experience? We believe that innovation efforts could be certainly addressed to improve usability and to satisfy all social needs that the travel experience generates. THINPLE can do a lot in many directions. Let's assume that D is a monument with NFC tag, relevant contents stored into the DB and a node created in the THINPLE graph. When A taps on the D's tag, s/he gets direct access to D's contents and, at the same time, D shows up in the A's social space thus enabling A to share with friends contents associated to D as well as to join the D's community. The main benefits achieved by using the THINPLE approach with monuments are the following:

- Usability - in this case, tapping on the NFC tag gives direct access to information relevant to the object a person has in front. A sort of autonomic

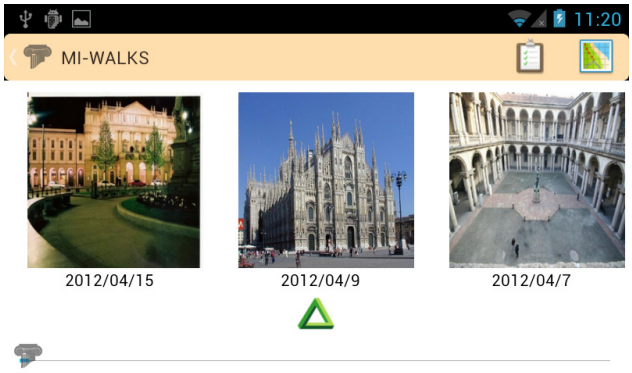


Fig. 4.3: *Timeline of encountered monuments.*

content retrieval directed by pointing the object in front of you. A natural and simple gesture.

- Human sociality - the travel is a life experience with known impact on the sociality of a person. THINPLE helps users to automatically enrich their social capital with contents about objects and friends, avoiding them using different apps to publish life experiences. THINPLE can be useful to share these experiences, to stimulate friends to do the same and to freely discuss about something you like.
- Diary - THINPLE automatically generates the user travel diary mirroring the real encounters (things and people) happening during user travel.
- Statistical analysis - All contact events with D are stored in the THINPLE-cloud and provide the institution promoting a cultural legacy with a rich dataset for operating its monuments and the communication channels to promote them.

4.4.2 Product traceability and control

The information system of the manufacturer of a domestic boiler can perfectly trace all the components that have been assembled to produce the final product, but it is generally unable to track the last hop to the customer which installs it at home. This makes difficult, for instance, to manage product recalls and to assist the customer for maintenance. Should the customer tap on the product tag when it is delivered at home, the product immediately enters the owner's environment and the missing link is created. At this point, we can envision a step forward; in fact, by providing the appliance with some communication capability and by means of some synoptic panel the customer will be able to remotely control his own boiler. This latter is an example in which the exchange of public keys at contact time (see Sec. 4.3) could be needed to secure the communication between the two parties.

4.5 Summary

In this chapter we have focused on micro-range interactions mediate by NFC radio technology. We have exploited NFC radio contacts to build up an online social network, named THINPLE, which realizes the Internet of Things and People paradigm. In this social network persons and daily life objects are seamlessly integrated in a unified social graph which is the expression of the real offline sociality of the users. We have used THINPLE to support a research study on the difference between online and offline human sociality and to support a mobile application to help tourists in keeping track of encountered monuments and point of interests.

Chapter 5

Conclusion

In this thesis we have proposed novel network solutions, mobile services and applications which are consistent with the next generation of mobile network vision. The 5th-Generation of mobile network will comprise multiple wireless technologies seamlessly integrated to offer to users a wide set of services and to guarantee them suitable QoE as well as ubiquitous and pervasive network access. In this work we have started from the users perspective by considering mobile users carrying multiple equipped devices with which they can interact at different spatial granularities. We have considered three radio technologies: cellular network, Wi-Fi, and NFC. We have shown that a clear understanding of characteristic of users interactions can support the designing of new network architectures, the provisioning of network resources, and the development of novel mobile services and applications.

We have analyzed long-range interactions mediated by cellular network by means of two mobile datasets. By using data mining techniques, we have studied different aspects of mobile users behavior such as mobility, burstiness, and sociality. The analysis of the mobility aspects has shown that users exhibit very regular mobility patterns visiting a small set of favorite locations. Moreover, we have analyzed the impact of mobility on phone interactions and we found that a considerable amount of phone calls (around 25%) are performed between persons who are in proximity to each other. Based on these findings, we have proposed a novel LTE-based network architecture placed on the edge network and able to support in-proximity communication channels and local data traffic offload. As a proof of concept, we have considered a newspaper/magazine distribution service. We have performed extensive simulations using real mobility traces extracted from the dataset and we have shown the benefits of the proposed network architecture from both operator and user points of view. We then analyzed the temporal inhomogeneities of mobile users behavior and we have found that users are usually organizing their phone activities in bursts. We have extended previous literature results by considering the multidimensionality of mobile users phone activities. We have developed a theoretical framework suitable to describe the interplay be-

tween different communication media (calls and text messages in our case). By applying this framework we have found that mobile users have the tendency to organize their phone activities in order to minimize the switches between different communication channels, rather than follow some order induced by their sociality. Furthermore, we have developed a burst generative model based on GPSN, which is able to correctly reproduce the observed interplay between the communication media. Last, we have analyzed how mobile users manage their sociality through mobile phones. We found that users interact mostly with a small subset of personal contacts, confirming the Dunbar grooming network theory. Based on this analysis we have developed a new mobile application which helps users to manage their sociality by means of a mobile phone and to keep track of the temporal evolution of their grooming network.

As a next step, we have considered medium- and short-range interactions mediated by Wi-Fi radio technology. We have exploited this kind of interactions to create an Opportunistic Network (ON) deployed on top of Public Transportation System (PTS) suitable to support delay tolerant applications and services. By means of extensive simulations on the actual PTS of the city of Milan we have shown that the single copy routing algorithm Op-HOP scales at urban level avoiding the overhead of multiple replicas which affects MaxProp and RAPID. Then, we have addressed the allocation problem for a network infrastructure to augment the overall ON performance. The proposed solution consists in the creation of a road-side network infrastructure by means of disconnected relays we call Road-Side eXchange Points (RSXP). Using a suitable algorithm, we identify a proper subset of bus stops candidate to take part in fixed infrastructure. By means of extensive simulations we have shown the impact of RSXPs density on the DTN performances. Results indicate that delivery delay will benefit from the presence of the infrastructure but, in order to improve the delivery ratio, a minimum density is required. The number of traversed hops indicates that the infrastructure involvement in the forwarding process is upper bounded and deployment of RSXPs above a certain density is a potential waste of resources. Using these results, we have demonstrated that a lightweight, and non invasive, road-side infrastructure is a feasible way to deploy a DTN over a PTS to provide mobile applications and services based on location and context on a urban scale.

Finally, we have focused on micro-range interactions mediate by NFC radio technology. We have exploited NFC radio contacts to build up an online social network, named THINPLE, which realizes the Internet of Things and People paradigm. In this social network persons and daily life objects are seamlessly integrated in a unified social graph which is the expression of the real offline sociality of the users. We have used THINPLE to support a research study on the difference between online and offline human sociality and to support a mobile application to help tourists in keeping track of encountered monuments and point of interests.

Appendix A

URBeS: Urban Backbone Routing Simulator

URBeS (Urban Routing Backbone Simulator) is the experimental platform we developed which is able to evaluate the QoS performance of a city BSN. This is possible by acquiring real city topological data as well as the relative PTS timetable. URBeS is able to accurately reproduce bus movements in real urban environments and to simulate data forwarding among buses. Our experimental platform is able to support any external routing policy in order to compare performance of various routing algorithms. The functional scheme of URBeS, reported in Fig. A.1, aims to emphasize both the modular design of its functionalities and its support for any externally specified routing policy.

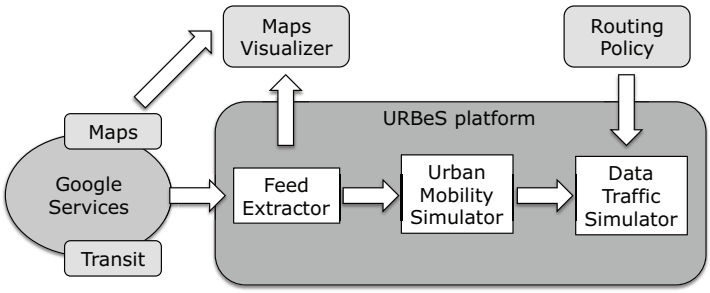


Fig. A.1: *URBeS functional scheme.*

The analysis of a city PTS starts from a Google transit [23] feed, which is a database of planned trips, provided by a transit authority from which we devise a movement model. GPS coordinates of every bus stop have later been converted to Cartesian coordinates. Nodes along each line move between coordinates accordingly to real timetables moving at a constant speed. Pauses at stops are – when planned – included in the transit feed and thus simulated accordingly.

The URBeS framework is then composed of three sequential modules.

In the first phase, URBeS parses information from the feed and produces a timetable of bus movements together with a topology of the PTS layout.

The output from the first phase is fed to an urban mobility simulation module which is in charge of generate mobility traces for all the buses based on the real PTS timetable. In this phase, URBeS also computes statistics about bus contacts. They are useful for predicting intra- and inter-contact times and for understanding city coverage of the PTS.

The last phase adds data traffic to the picture: network traffic is generated randomly by each bus and is delivered following the provided routing policy. URBeS logs detailed information on profile delivery rates, delays, and locations where forwarding takes place. Simulation is performed by means of a custom-made discrete event simulation software we developed around the concept of BSN. This development has been required in order to overcome poor scalability in term of total traffic and number of nodes from existing products (e.g., GloMoSim [112] and ns-2[1]), while introducing an highly optimized urban canyon model which fails to be even in more modern simulation environments (e.g., the ONE simulator [67]). Results from our simulator have been positively evaluated against GloMoSim for low traffic levels using the same output coming from Urban Mobility Simulator. Our simulator can thus provide a valid comparison between different routing policies, while allowing easy testing on multiple urban environments.

In the following sections we discuss the implementation details relative to the three components of URBeS.

A.1 Google transit Feed Extractor

The Google transit Feed Extractor (FE) is actually a data converter tool which reads a feed and stores all data useful for simulation in a more manageable format.

A Google transit feed is a database comprised by several tables reporting individual bus trips. Each trip is classified by a time context (e.g., Monday, Sunday, or "New Years Eve"), a route id, and a bus head sign. In order to build the topology, the FE starts from the list of trips and classifies all possible paths based on the stops made. Paths are then grouped together with reversals and aliases; reversals are paths making the same stops in reverse order, whereas aliases are routes having the same end of the line and almost the same intermediate stops (i.e., their difference is below a certain threshold). All paths in a group are identified as a single – looping – line, as traditionally understood by passengers. All lines must be closed paths as that way it is possible to assign buses for subsequent departures once they have completed a round trip.

Lines which does not define a closed path are not used for data transportation because it is not possible for us to make a reasonable assumptions about

[1] ns-2 simulator is available at `http://nsnam.isi.edu/nsnam/`

the future activity of buses running along them (such as in the case of shuttles making one-way trips from the train station to the airport where there is no timetable available for the way back). Thus, following the normal procedures, we would be required to consider a bus reaching the end of the line as to be going out of service and discarding all transported packets. Needless to say, such a behavior is not useful for routing where encounters are not deterministic and, as a consequence, such lines are not considered to be part of the data distribution system.

The output generated from the FE is a database of routes and timetables from real traces which are fed to the second module of URBeS.

A.2 Urban Mobility Simulator

The Urban Mobility Simulator (UMS) generates mobility traces for all buses during a given period by using the database created by the FE.

The UMS module creates bus instances as needed to ensure scheduled starts and manages buses going out of services when the PTS is overpopulated. Buses leave the line head at the scheduled time and make all the stops according to the set timetable; while completing a path a bus waits at the head of the line, if necessary, for the next scheduled departure. If there are already two buses in line, then we consider it to be overpopulated and the bus goes out of service. In case of aliases, the bus waiting in line will select the path of the next aliased line departing from the current head of line.

Since location and timetables from Google Transit are driving nodes during simulation, UMS is completely decoupled from a movement model: we have deterministic information about system evolution and there is no need to add parameters coming from any external model. Give that in a real PTS, timetables are planned taking into account average daily traffic, along the day we are not required to predict and simulate traffic jams. Nevertheless, we added random noise of up to ten minutes on the scheduled departure times, in order to take into consideration small variations between average traffic and actual street condition. Moreover, to determine if two buses are in radio contact, a line-of-sight model is adopted. Urban canyons play a fundamental role in wireless links and we take them into account by using a street map created from bus routes.

At this stage, we compute a first analysis based on topology and bus movement. The analysis provides the evolution of bus populations and relative contacts, information about the number of neighbors, the and distribution of intra- and inter-contact times.

The output of the UMS is a trace of bus movements in a bi-dimensional space.

A.3 Data Traffic Simulator

The Data Traffic Simulator (DTS) leverages the output of UMS by introducing network traffic and applying a routing policy.

When a packet is generated it is placed in the node local buffer until forwarding becomes possible; an isolated bus will keep accumulating packets while no contacts are experienced. When an encounter happens all the packets are checked for forwarding. Forwarding is subject to bandwidth and buffer limitations. Bandwidth is accounted using a token bucket mechanism while buffer space availability is simply checked before transmission. In case of contention a first-in-first-out policy is applied. Packets are then forwarded in accordance with the adopted routing policy.

When a bus reaches the end of the line it may or may not queue up and wait for another scheduled departure. If the bus keeps a place in line it will hold all its data and will keep generating packets while waiting. If, on the other hand, the bus leaves service all content will be pushed to the first bus waiting in line. If there is no bus – because there are no more scheduled departures – all the stored packets are dropped and considered lost.

The routing policy used by the BSN is provided as an external module and may be either link-state or distance-vector. A user may implement his/her own routing policy by defining the application logic based on which forwarding will take place. This approach makes URBeS an ideal platform for routing protocol development and comparison: multiple experiments can be easily run specifying different routing modules to be used with the same traffic pattern.

The output of the DTS is a complete trace of the generated data traffic. From this trace we can compute all needed performance indexes and perform protocol analysis.

References

1. 3GPP. Local ip access and selected ip traffic offload (lipa-sipto) (rel-10). Technical report, 3GPP TS 23.829, 2011.
2. S. Ahmed and S.S. Kanhere. Cluster-based forwarding in delay tolerant public transport networks. In *Local Computer Networks, 2007. LCN 2007. 32nd IEEE Conference on*, pages 625 –634, Oct. 2007.
3. Marco Ajmone Marsan, Gianni Conte, and Gianfranco Balbo. A Class of Generalized Stochastic Petri Nets for the Performance Evaluation of Multiprocessor Systems. *ACM Transaction on Computer Systems*, 2(2):93–122, May 1984.
4. O.G. Aliu, A Imran, M.A Imran, and B. Evans. A survey of self organisation in future cellular networks. *Communications Surveys Tutorials, IEEE*, 15(1):336–361, First 2013.
5. Arash Asadi and Vincenzo Mancuso. Wifi direct and LTE D2D in action. In *Proceedings of the IFIP Wireless Days, WD 2013, Valencia, Spain, November 13-15, 2013*, pages 1–8, 2013.
6. Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer Networks*, 54(15):2787 – 2805, 2010.
7. Aruna Balasubramanian, Brian Levine, and Arun Venkataramani. Dtn routing as a resource allocation problem. In *Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '07, pages 373–384, New York, NY, USA, 2007. ACM.
8. Nilanjan Banerjee, Mark D. Corner, and Brian Neil Levine. Design and Field Experimentation of an Energy-Efficient Architecture for DTN Throwboxes. *IEEE/ACM Transactions on Networking*, 18(2):554–567, April 2010.
9. B. Bangerter, S. Talwar, R. Arefi, and K. Stewart. Networks and devices for the 5g era. *Communications Magazine, IEEE*, 52(2):90–96, February 2014.
10. Albert-Laszlo Barabasi. The origin of bursts and heavy tails in human dynamics. *Nature*, 435(7039):207–211, 05 2005.
11. Michele Berlingerio, Michele Coscia, Fosca Giannotti, Anna Monreale, and Dino Pedreschi. Multidimensional networks: foundations of structural analysis. *World Wide Web*, pages 1–27, 2012.
12. Sourjya Bhaumik, Shoban Preeth Chandrabose, Manjunath Kashyap Jataprolu, Gautam Kumar, Anand Muralidhar, Paul Polakos, Vikram Srinivasan, and Thomas Woo. Cloudiq: A framework for processing base stations in a data center. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, Mobicom '12, pages 125–136, New York, NY, USA, 2012. ACM.

13. Piotr Bródka, Przemysław Kazienko, Katarzyna Musiał, and Krzysztof Skibicki. Analysis of neighbourhoods in multi-layered dynamic social networks. *International Journal of Computational Intelligence Systems*, 5(3):582–596, 2012.
14. J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine. Maxprop: Routing for vehicle-based disruption-tolerant networks. pages 1 –11, April 2006.
15. Han Cai and Do Young Eun. Aging rules: what does the past tell about the future in mobile ad-hoc networks? In *MobiHoc '09: Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, pages 115–124. ACM, 2009.
16. Francesco Calabrese, Zbigniew Smoreda, Vincent D. Blondel, and Carlo Ratti. Interplay between telecommunications and face-to-face interactions: A study using mobile phone data. *PLoS ONE*, 6(7):e20814, 07 2011.
17. Julián Candia, Marta C González, Pu Wang, Timothy Schoenharl, Greg Madey, and Albert-László Barabási. Uncovering individual and collective human dynamics from mobile phone records. *Journal of Physics A: Mathematical and Theoretical*, 41(22):224015, 2008.
18. T Trevor Caughlin, Nick Ruktanonchai, Miguel A Acevedo, Kenneth K Lopiano, Olivia Prosper, Nathan Eagle, and Andrew J Tatem. Place-based attributes predict community membership in a mobile phone communication network. *PloS one*, 8(2):e56057, 2013.
19. Chia-Hsuan Chang, Chia-Lung Liu, Hsi-Lu Chao, Kuei-Li Huang, and Yi-Bing Lin. A novel lipa scheme for lte voip services with home enbs. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, 4(3):1–22, 9 2013.
20. G. Chiola, M. A. Marsan, G. Balbo, and G. Conte. Generalized stochastic petri nets: A definition at the net level and its implications. *IEEE Trans. Softw. Eng.*, 19(2):89–107, February 1993.
21. Cisco. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2013-2018. Technical report, 2013.
22. G.B. Colombo, M.J. Chorley, M.J. Williams, S.M. Allen, and R.M. Whitaker. You are where you eat: Foursquare checkins as indicators of human mobility and behaviour. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on*, 2012.
23. Balázs Cs. Csáji, Arnaud Browet, V.A. Traag, Jean-Charles Delvenne, Etienne Huens, Paul Van Dooren, Zbigniew Smoreda, and Vincent D. Blondel. Exploring the mobility of mobile phone users. *Physica A: Statistical Mechanics and its Applications*, 392(6):1459 – 1473, 2013.
24. A Damnjanovic, J. Montojo, Yongbin Wei, Tingfang Ji, Tao Luo, M. Vajapeyam, Taesang Yoo, Osok Song, and D. Malladi. A survey on 3gpp heterogeneous networks. *Wireless Communications, IEEE*, 18(3):10–21, June 2011.
25. Carina T. De Oliveira, Reinaldo B. Braga, Danilo M. Taveira, Natalia C. Fern, and Otto Carlos M. B. Duarte. A predicted-contact routing scheme for brazilian rural networks. In *Electrical Engineering Program, COPPE/UFRJ*, 2008.
26. Michael Demmer and Kevin Fall. Dtlsr: delay tolerant routing for developing regions. In *NSDR '07: Proceedings of the 2007 workshop on Networked systems for developing regions*, pages 1–6. ACM, 2007.
27. Ben Dodson, Ian Vo, T.J. Purtell, Aemon Cannon, and Monica Lam. Musubi: disintermediated interactive social feeds for mobile devices. In *Proceedings*

of the 21st international conference on World Wide Web, WWW '12, pages 211–220, New York, NY, USA, 2012. ACM.
28. R.I.M. Dunbar and M. Spoors. Social networks, support cliques, and kinship. *Human Nature*, 6(3):273–290, 1995.
29. N. Eagle, A. Pentland, and D. Lazer. Inferring friendship network structure by using mobile phone data. *Proceedings of The National Academy of Sciences*, 106:15274–15278, 2009.
30. Nathan Eagle and Alex (Sandy) Pentland. Reality mining: Sensing complex social systems. *Personal Ubiquitous Comput.*, 10(4):255–268, March 2006.
31. Jean-Pierre Eckmann, Elisha Moses, and Danilo Sergi. Entropy of dialogues creates coherent structures in e-mail traffic. *Proceedings of the National Academy of Sciences of the United States of America*, 101(40):14333–14337, 2004.
32. Ericsson. 5g radio access. White Paper, June 2013.
33. Ericsson. The real-time cloud. White Paper, February 2014.
34. Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining*, volume 96 of *KDD '96*, pages 226–231, 1996.
35. Kevin Fall. A delay-tolerant network architecture for challenged internets. In *SIGCOMM '03: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 27–34. ACM, 2003.
36. S. Gaito, D. Maggiorini, C. Quadri, and G.P. Rossi. Selective offload and proactive caching of mobile data in lte-based urban networks. In *Mobile Data Management (MDM), 2013 IEEE 14th International Conference on*, volume 1, pages 271–274, June 2013.
37. S. Gaito, D. Maggiorini, G.P. Rossi, and A. Sala. Bus switched networks: An ad hoc mobile platform enabling urban-wide communications. *Ad Hoc Networks*, 10(6):931 – 945, 2012.
38. S. Gaito, E. Pagani, and G. P. Rossi. Strangers help friends to communicate in opportunistic networks. *Computer Networks*, 55(2):374–385, 2011.
39. S. Gaito, E. Pagani, G. P. Rossi, and M. Zignani. Sensing multi-dimensional human behavior in opportunistic networks. In *MobiOpp'12 - Proceedings of the 3rd ACM International Workshop on Mobile Opportunistic Networks*, pages 89–90, 2012.
40. S. Gaito, G.P. Rossi, and M. Zignani. Facencounter: Bridging the gap between offline and online social networks. In *Signal Image Technology and Internet Based Systems (SITIS), 2012 Eighth International Conference on*, pages 768–775, Nov 2012.
41. Sabrina Gaito, Dario Maggiorini, Christian Quadri, and Gian Paolo Rossi. On the impact of a road-side infrastructure for a dtn deployed on a public transportation system. In *Networking (2)*, pages 265–276, 2012.
42. Sabrina Gaito, Giovanni Manta, Christian Quadri, Gian Paolo Rossi, and Matteo Zignani. Groo-me: Handling the dynamics of our sociality on mobile phone. In *Wireless and Mobile Networking Conference (WMNC), 2014 7th IFIP*, pages 1–4. IEEE, 2014.
43. Sabrina Gaito, Christian Quadri, Gian Paolo Rossi, and Matteo Zignani. Thinple - the new online sociality is built on top of nfc-based contacts. In *Wireless Days*, pages 1–5, 2012.

44. Sabrina Gaito, Matteo Zignani, Gian Paolo Rossi, Alessandra Sala, Xiaohan Zhao, Haitao Zheng, and Ben Y. Zhao. On the bursty evolution of online social networks. In *Proceedings of the First ACM International Workshop on Hot Topics on Interdisciplinary Social Networks Research*, HotSocial '12, pages 1–8, New York, NY, USA, 2012. ACM.
45. Marta C. Gonzalez, Cesar A. Hidalgo, and Albert-Laszlo Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, 2008.
46. Aditya Gudipati, Daniel Perry, Li Erran Li, and Sachin Katti. Softran: Software defined radio access network. In *Proceedings of the Second ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking*, HotSDN '13, pages 25–30, New York, NY, USA, 2013. ACM.
47. Bo Han, Pan Hui, V.S.A. Kumar, M.V. Marathe, Jianhua Shao, and A. Srinivasan. Mobile data offloading through opportunistic communications and social participation. *Mobile Computing, IEEE Transactions on*, 11(5):821 –834, may 2012.
48. Leo Den Hartog, Thrasyvoulos Spyropoulos, and Franck Legendre. Using public transportation as a dtn backbone: Mobility properties and performance analysis. In *Proceedings of AOC 2010*, Montreal, Canada, 2010. IEEE.
49. Cesar A. Hidalgo and C. Rodriguez-Sickert. The dynamics of a mobile phone network. *Physica A: Statistical Mechanics and its Applications*, 387(12):3017 – 3024, 2008.
50. Petter Holme. Network dynamics of ongoing social relationships. *EPL (Europhysics Letters)*, 64(3):427, 2003.
51. Theus Hossmann, Franck Legendre, George Nomikos, and Thrasyvoulos Spyropoulos. Stumbl: Using facebook to collect rich datasets for opportunistic networking research. In *WOWMOM*, pages 1–6, 2011.
52. Wei-jen Hsu, Debojyoti Dutta, and Ahmed Helmy. Profile-cast: Behavior-aware mobile networking. *SIGMOBILE Mob. Comput. Commun. Rev.*, 12(1):52–54, January 2008.
53. Pan Hui, Augustin Chaintreau, James Scott, Richard Gass, Jon Crowcroft, and Christophe Diot. Pocket switched networks and human mobility in conference environments. In *WDTN '05: Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*, pages 244–251. ACM, 2005.
54. Pan Hui, Jon Crowcroft, and Eiko Yoneki. Bubble rap: social-based forwarding in delay tolerant networks. In *MobiHoc '08: Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, pages 241–250. ACM, 2008.
55. Sushant Jain, Kevin Fall, and Rabin Patra. Routing in a delay tolerant network. In *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 145–158. ACM, 2004.
56. Wei jen Hsu, Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Ahmed Helmy. Modeling time-variant user mobility in wireless mobile networks. In *INFOCOM*, 2007.
57. J.G. Jetcheva, Y.-C. Hu, S. PalChaudhuri, A.K. Saha, and D.B. Johnson. Design and evaluation of a metropolitan area multitier wireless ad hoc network architecture. In *Mobile Computing Systems and Applications, 2003.*, pages 32 – 43, Oct. 2003.

58. Zhi-Qiang Jiang, Wen-Jie Xie, Ming-Xia Li, Boris Podobnik, Wei-Xing Zhou, and H. Eugene Stanley. Calling patterns in human communication dynamics. *Proceedings of the National Academy of Sciences*, 110(5):1600–1605, 2013.
59. Xin Jin, Li Erran Li, Laurent Vanbever, and Jennifer Rexford. Softcell: Scalable and flexible cellular core network architecture. In *Proceedings of the Ninth ACM CoNEXT 2013*, pages 163–174, New York, NY, USA, 2013. ACM.
60. Yu Jin, Nick Duffield, Alexandre Gerber, Patrick Haffner, Wen-Ling Hsu, Guy Jacobson, Subhabrata Sen, Shobha Venkataraman, and Zhi-Li Zhang. Characterizing data usage patterns in a large cellular network. In *Proceedings of the 2012 ACM SIGCOMM Workshop on Cellular Networks: Operations, Challenges, and Future Design*, CellNet '12, pages 7–12, New York, NY, USA, 2012. ACM.
61. Hang-Hyun Jo, Márton Karsai, Juuso Karikoski, and Kimmo Kaski. Spatiotemporal correlations of handset-based service usages. *EPJ Data Science*, 1(1):1–18, 2012.
62. Hang-Hyun Jo, Márton Karsai, János Kertész, and Kimmo Kaski. Circadian pattern and burstiness in mobile phone communication. *New Journal of Physics*, 14(1):013055, 2012.
63. Hang-Hyun Jo, Raj Kumar Pan, and Kimmo Kaski. Time-varying priority queuing models for human dynamics. *Phys. Rev. E*, 85:066101, Jun 2012.
64. Hang-Hyun Jo, Raj Kumar Pan, Juan I. Perotti, and Kimmo Kaski. Contextual analysis framework for bursty dynamics. *Phys. Rev. E*, 87:062131, Jun 2013.
65. Márton Karsai, Kimmo Kaski, Albert-László Barabási, and János Kertész. Universal features of correlated bursty behaviour. *Sci. Rep.*, 2, 05 2012.
66. Márton Karsai, Kimmo Kaski, and János Kertész. Correlated dynamics in egocentric communication networks. *PLoS ONE*, 7(7):e40612, 07 2012.
67. Ari Keränen, Jörg Ott, and Teemu Kärkkäinen. The ONE Simulator for DTN Protocol Evaluation. In *SIMUTools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, New York, NY, USA, 2009. ICST.
68. Riivo Kikas, Marlon Dumas, and Márton Karsai. Bursty egocentric network evolution in skype. *Social Network Analysis and Mining*, pages 1–9, 2013.
69. Jon Kleinberg. Bursty and hierarchical structure in streams. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 91–101, New York, NY, USA, 2002. ACM.
70. Hans-Peter Kriegel, Peer Kröger, Jörg Sander, and Arthur Zimek. Density-based clustering. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(3):231–240, 2011.
71. Renaud Lambiotte, Lionel Tabourier, and Jean-Charles Delvenne. Burstiness and spreading on temporal networks. *Arxiv preprint 1305.0543*, 2013.
72. Jure Leskovec and Eric Horvitz. Planetary-scale views on a large instant-messaging network. In *Proceedings of the 17th international conference on World Wide Web*, WWW'08, pages 915–924. ACM, 2008.
73. Anders Lindgren, Avri Doria, and Olov Schelén. Probabilistic routing in intermittently connected networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7(3):19–20, 2003.
74. Cong Liu and Jie Wu. An optimal probabilistic forwarding protocolin delay tolerant networks. In *MobiHoc '09: Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, pages 105–114. ACM, 2009.

75. Dario Maggiorini, Christian Quadri, and LauraAnna Ripamonti. Opportunistic mobile games using public transportation systems: a deployability study. *Multimedia Systems*, pages 1–18, 2014.
76. Mary McGlohon et al. Finding patterns in blog shapes and blog evolution. In *Proc. of ICWSM*, 2007.
77. Giovanna Miritello, Esteban Moro, and Rubén Lara. Dynamical strength of social ties in information spreading. *Physical Review E*, 83:045102, Apr 2011.
78. Giovanna Miritello, Esteban Moro, Rubén Lara, Rocío Martínez-López, John Belchamber, Sam G. B. Roberts, and Robin I. M. Dunbar. Time as a limited resource: Communication strategy in mobile phone networks. *Social Networks*, 35(1):89–95, 2013.
79. Peter J. Mucha, Thomas Richardson, Kevin Macon, Mason A. Porter, and Jukka-Pekka Onnela. Community structure in time-dependent, multiscale, and multiplex networks. *Science*, 328(5980):876–878, 2010.
80. Amit Anil Nanavati, Rahul Singh, Dipanjan Chakraborty, Koustuv Dasgupta, Sougata Mukherjea, Gautam Das, Siva Gurumurthy, and Anupam Joshi. Analyzing the structure and evolution of massive telecom graphs. *Knowledge and Data Engineering, IEEE Transactions on*, 20(5):703–718, 2008.
81. S.C. Nelson, M. Bakht, and R. Kravets. Encounter-based routing in dtns. In *INFOCOM 2009, IEEE*, pages 846 –854, April 2009.
82. J.-P. Onnela, J. Saramäki, J. Hyvönen, G. Szabó, D. Lazer, K. Kaski, J. Kertész, and A.-L. Barabási. Structure and tie strengths in mobile communication networks. *Proceedings of the National Academy of Sciences*, 104(18):7332–7336, 2007.
83. J.-P. Onnela, J. Saramäki, J. Hyvönen, G. Szabó, D. Lazer, K. Kaski, J. Kertész, and A.-L. Barabási. Structure and tie strengths in mobile communication networks. *Proceedings of the National Academy of Sciences*, 104(18):7332–7336, 2007.
84. G. Palla, A. Barabasi, and T. Vicsek. Quantifying social group evolution. *Nature*, 446:664–667, 2007.
85. M. Papandrea, M. Zignani, S. Gaito, S. Giordano, and G.P. Rossi. How many places do you visit a day? In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on*, pages 218–223, March 2013.
86. U. Paul, AP. Subramanian, M.M. Buddhikot, and S.R. Das. Understanding traffic dynamics in cellular data networks. In *INFOCOM, 2011 Proceedings IEEE*, pages 882–890, April 2011.
87. A. Pentland, R. Fletcher, and A. Hasson. Daknet: rethinking connectivity in developing nations. *Computer*, 37(1):78 – 83, Jan. 2004.
88. Santi Phithakkitnukoon, Zbigniew Smoreda, and Patrick Olivier. Socio-geography of human mobility: A study using longitudinal mobile phone data. *PLoS ONE*, 7(6):e39253, 06 2012.
89. A-K Pietil"ainen and C. Diot. Dissemination in opportunistic social networks: The role of temporal communities: the role of temporal communities. In *MobiHoc'12: Proceedings of the Thirteenth International Symposium on Mobile Ad Hoc Networking and Computing*, June 2012.
90. Mikko Pitkanen, Andrea Passarella, Silvia Giordiano, Franck Legendre, Thrasyvoulos Spyropoulos, Joerg Ott, Teemu Karkkainen, Marco Conti,

Daniele Puccinelli, Martin May, Nidhi Hegde, Karin Hummel, and Sacha Trifunovic. Scampi: Service platform for social aware mobile and pervasive computing. In *Proc. ACM Mobile Cloud Computing Workshop*, 2012.

91. C. Quadri, D. Maggiorini, S. Gaito, and G. P. Rossi. On the scalability of delay-tolerant routing protocols in urban environment. In *IFIP Wireless Days*, volume 1, pages 1 –6, 2011.
92. Christian Quadri, Dario Maggiorini, Sabrina Gaito, and Gian Paolo Rossi. On the scalability of delay-tolerant routing protocols in urban environment. In *Wireless Days*, pages 1–6, 2011.
93. Christian Quadri, Matteo Zignani, Lorenzo Capra, Sabrina Gaito, and Gian Paolo Rossi. Multidimensional human dynamics in mobile phone communications. *PLoS ONE*, 9(7):e103183, 07 2014.
94. 3GPP TR 23.829. Local IP Access and Selected IP Traffic Offload (LIPA-SIPTO) (Release 10). Technical Report, 2011.
95. 3GPP TR 36.877. LTE Device to Device Proximity Services; User Equipment (UE) radio transmission and reception (Release 12). Technical Report, 2014.
96. Gabriel Sandulescu and Simin Nadjm-Tehrani. Opportunistic dtn routing with window-aware adaptive replication. In *AINTEC '08: Proceedings of the 4th Asian Conference on Internet Engineering*, pages 103–112. ACM, 2008.
97. C.B. Sankaran. Data offloading techniques in 3gpp rel-10 networks: A tutorial. *Communications Magazine, IEEE*, 50(6):46–53, June 2012.
98. M. Sede, Li Xu, Li Da, Wu Min-You, Li Minglu, and Shu Wei. Routing in large-scale buses ad hoc networks. In *Wireless Communications and Networking Conference, 2008.*, pages 2711 –2716, 2008.
99. Muhammad Zubair Shafiq, Lusheng Ji, Alex X. Liu, Jeffrey Pang, Shobha Venkataraman, and Jia Wang. A first look at cellular network performance during crowded events. In *Proceedings of the ACM SIGMETRICS 2013*, SIGMETRICS '13, pages 17–28, New York, NY, USA, 2013. ACM.
100. M.Z. Shafiq, Lusheng Ji, AX. Liu, J. Pang, and J. Wang. Characterizing geospatial dynamics of application usage in a 3g cellular data network. In *INFOCOM, 2012 Proceedings IEEE*, pages 1341–1349, March 2012.
101. Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S. Raghavendra. Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-tolerant Networking*, WDTN '05, pages 252–259, New York, NY, USA, 2005. ACM.
102. Michael Szell, Renaud Lambiotte, and Stefan Thurner. Multirelational organization of large-scale social networks in an online world. *Proceedings of the National Academy of Sciences*, 107(31), 2010.
103. Xenofon Vasilakos, Vasilios A. Siris, George C. Polyzos, and Marios Pomonis. Proactive selective neighbor caching for enhancing mobility support in information-centric networks. In *Proceedings of the second edition of the ICN workshop on Information-centric networking*, ICN '12, pages 61–66, New York, NY, USA, 2012. ACM.
104. Alexei Vázquez, João Gama Oliveira, Zoltán Dezsö, Kwang-Il Goh, Imre Kondor, and Albert-László Barabási. Modeling bursts and heavy tails in human dynamics. *Physical Review E*, 73, 2006.
105. Bimal Viswanath, Alan Mislove, Meeyoung Cha, and Krishna P. Gummadi. On the evolution of user interaction in facebook. In *Proceedings of the 2Nd ACM*

Workshop on Online Social Networks, WOSN '09, pages 37–42, New York, NY, USA, 2009. ACM.

106. Dashun Wang, Dino Pedreschi, Chaoming Song, Fosca Giannotti, and Albert-Laszlo Barabasi. Human mobility, social ties, and link prediction. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 1100–1108, New York, NY, USA, 2011. ACM.
107. Christo Wilson, Alessandra Sala, Krishna P. N. Puttaswamy, and Ben Y. Zhao. Beyond social graphs: User interactions in online social networks and their implications. *ACM Transactions on the Web*, 6(4), 2012.
108. Ye Wu, Changsong Zhou, Jinghua Xiao, Jürgen Kurths, and Hans Joachim Schellnhuber. Evidence for a bimodal distribution in human communication. *Proceedings of the National Academy of Sciences*, 2010.
109. Ye Wu, Changsong Zhou, Jinghua Xiao, Jürgen Kurths, and Hans Joachim Schellnhuber. Evidence for a bimodal distribution in human communication. *Proceedings of the National Academy of Sciences*, 2010.
110. Qiang Xu, Junxian Huang, Zhaoguang Wang, Feng Qian, Alexandre Gerber, and Zhuoqing Morley Mao. Cellular data network infrastructure characterization and implication on mobile content placement. In *Proceedings of the ACM SIGMETRICS Joint International Conference on Measurement and Modeling of Computer Systems*, SIGMETRICS '11, pages 317–328, New York, NY, USA, 2011. ACM.
111. Quan Yuan, Ionut Cardei, and Jie Wu. Predict and relay: an efficient routing in disruption-tolerant networks. In *MobiHoc '09: Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, pages 95–104. ACM, 2009.
112. X. Zeng, R. Bagrodia, and M. Gerla. Glomosim: a library for parallel simulation of large-scale wireless networks. In *Parallel and Distributed Simulation, 1998. PADS 98. Proceedings. Twelfth Workshop on*, pages 154–161, May 1998.
113. Xiaolan Zhang, Jim Kurose, Brian Neil Levine, Don Towsley, and Honggang Zhang. Study of a bus-based disruption-tolerant network: mobility modeling and impact on routing. In *MobiCom '07: Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, pages 195–206. ACM, 2007.
114. Xin Zhang, D. Shasha, Yang Song, and J.T.-L. Wang. Fast elastic peak detection for mass spectrometry data mining. *Knowledge and Data Engineering, IEEE Transactions on*, 24(4):634–648, 2012.
115. Wenrui Zhao, Yang Chen, Mostafa Ammar, Mark Corner, Brian Levine, and Ellen Zegura. Capacity enhancement using throwboxes in dtns. In *In Proc. IEEE Intl Conf on Mobile Ad hoc and Sensor Systems (MASS*, pages 31–40, 2006.
116. M. Zignani and S. Gaito. Extracting human mobility patterns from gps-based traces. In *Wireless Days (WD), 2010 IFIP*, pages 1–5. IEEE, 2010.
117. M. Zignani, S. Gaito, and G. Rossi. Extracting human mobility and social behavior from location-aware traces. *Wireless Communications and Mobile Computing*, 2012.
118. Matteo Zignani, Christian Quadri, Sabrina Gaito, and Gian Paolo Rossi. Exploiting all phone media? a multidimensional network analysis of phone users' sociality. *Arxiv preprint*, 2013.

119. J.C. Zuniga, C.J. Bernardos, A De La Oliva, T. Melia, R. Costa, and A Reznik. Distributed mobility management: A standards landscape. *Communications Magazine, IEEE*, 51(3):80–87, March 2013.