Thèse de Doctorat de l'Université Paris VI
Pierre et Marie Curie

Spécialité
SYSTÈMES INFORMATIQUE

presentée par

Jérémie Leguay

pour obtenir le grade de

Docteur de l'Université Pierre et Marie Curie

Hétérogénéités et Routages dans les Réseaux à Connectivité Intermittente

A soutenir le 23 juillet 2007 devant le jury composé de:

M. :	Eric	FLEURY	Prof. à l'INSA de Lyon	Rapporteur
M. :	Martin	MAY	Chercheur senior associé à l'ETH Zürich	Rapporteur
M. :	Pierre	SENS	Prof. à l'Univ. Pierre & Marie Curie	Examinateur
M. :	Serge	FDIDA	Prof. à l'Univ. Pierre & Marie Curie	Directeur de Thèse
M. :	Timur	FRIEDMAN	Maître de Conf. à l'Univ. Pierre & Marie Curie	co-encadrant
M. :	Vania	CONAN	Ingénieur de recherche à Thales Communications	co-encadrant

Doctor of Science Thesis
Pierre & Marie Curie University (Paris VI)

Specialization
COMPUTER SCIENCE

presented by

Jérémie Leguay

submitted for the degree of

Doctor of Science of the Pierre & Marie Curie University

Heterogeneity and Routing in Delay Tolerant Networks

To be presented on July 2007, 23rd

Committee in charge:

M. :	Eric	FLEURY	Prof. at INSA - Lyon	Reviewer
M. :	Martin	MAY	Senior Research Associate - ETH Zürich	Reviewer
M. :	Pierre	SENS	Prof. - Univ. Pierre & Marie Curie	Examiner
M. :	Serge	FDIDA	Prof. - Univ. Pierre & Marie Curie	Advisor
M. :	Timur	FRIEDMAN	Assistant prof. - Univ. Pierre & Marie Curie	co-advisor
M. :	Vania	CONAN	Research engineer - Thales Communications	co-advisor

Résumé

Le domaine des réseaux tolérants aux délais (DTN) a réellement émergé ces deux dernières années afin de fournir des mécanismes permettant d'étendre l'architecture de l'Internet actuel. Les réseaux adressés par ce domaine partagent le fait que leur connectivité est perturbée ou que le niveau d'hétérogénéité est tel que les protocoles usuels de l'Internet ne fonctionnent plus. Nous avons étudié les problèmes liés au routage dans les réseaux connectés par intermittence.

Dans cette thèse, nous présentons plusieurs contributions pour le routage DTN. Celles-ci prennent place dans des réseaux où les entités participantes sont mobiles (e.g., téléphones, PDAs,...) et transportées par des gens ayant des liens sociaux (e.g., étudiants d'un même programme à l'université). D'abord, nous démontrons, à l'aide d'une analyse de traces réelles, la présence d'hétérogénéité dans les interactions entre les noeuds et que celle-ci peut être prise en compte pour proposer des stratégies de routage efficaces. Deuxièmement, nous proposons l'utilisation d'un formalisme générique basé sur un espace virtuel euclidien, appelé MobySpace, construit à partir d'informations sur les habitudes de mobilité des noeuds. Nous démontrons, avec le rejeux de traces de mobilité réelles, que ce formalisme peut s'appliquer au routage DTN et qu'il permet de créer des stratégies performantes en terme de taux de livraison et de coût de communication. Enfin, nous étudions la faisabilité d'une architecture de distribution de contenu en environnement urbain à l'aide de bornes courte portée Bluetooth. Nous étudions plusieurs stratégies de distribution en rejouant des traces que nous avons collectées lors d'une expérience inédite à Cambridge, GB.

Mots clés

routage, réseaux tolérants aux délais, connectivité intermittente

Abstract

There is a growing interest in Delay Tolerant Networking (DTN). This research field aims at providing communication means to extend the current Internet architecture for the support of *challenged networks*. These networks are mainly characterized by the fact that connectivity between entities suffers from disruptions. We examine in this thesis the problem of routing by using knowledge about network connectivity.

In this thesis, we make several contributions to DTN routing. We investigate particularly scenarios where network entities are mobile (e.g., mobile phones, PDAs) and carried by people sharing social relationships. First, we show, with the analysis of real traces, that there is heterogeneity in interactions between participants of such networks and we demonstrate that this heterogeneity could be taken into account to propose efficient routing schemes. Second, moving in such direction, we propose routing algorithms based on the use of a high-dimensional Euclidean space, that we call MobySpace, constructed upon nodes' mobility patterns. We have shown, through the replay of real mobility traces, that MobySpace-based routing schemes can applied to DTNs and that it can bring benefits in terms of enhanced bundle delivery and reduced communication costs. Finally, to contribute to the on-going data collection effort, we present an analysis of contact traces that we collected in an experiment we conducted in Cambridge, UK. This experiment allowed us to study the feasibility of a city-wide content distribution architecture composed of short range wireless access points.

Keywords

routing, delay tolerant networks, intermittent connectivity

Contents

Chapter 1

Introduction

NOWADAYS, networking technologies are at the center of computer systems. They communicate and benefit from the large amount of widespread networking capabilities introduced these last thirty years. As a consequence, more and more systems, having each their particularities and eventually their own communication facilities, are being interconnected through the Internet or through other kinds of networks. An example of such an interaction chain could be a mobile phone using a broadband access (e.g. UMTS or WiMax) to reach, through the Internet, a camera embedded in a geostationary satellite.

Mass market electronic devices such as laptops, PDA, music players, and mobile phones have also seen and continue to see a tremendous evolution, getting ever smaller and more present in users' lives. Users want to be mobile, to access remote content or personal information from wherever they are in a transparent manner. This is the concept of *pervasive* communications. Communication capabilities have thus increased to allow these devices to access content or services in infrastructure networks and to allow them to interact together.

Due to these changes in user needs, in availability of communication opportunities and in network heterogeneity, it is essential to develop new communication technologies and applications. Current networking approaches do not tolerate a high level of network heterogeneity, constrained environments, or connectivity disruptions caused by mobility or opportunistic communications. The growing number of devices that people carry, the variety of networking capabilities, the number of short and long range communication opportunities, and the heterogeneity of networks have lead researchers to envisage new ways to communicate. One class of solutions, Delay Tolerant Networking (DTN), can be view as a way to extend the Internet's networking capabilities. It offers the possibility for users to communicate when connectivity to an access network is intermittent or when the local network is composed only of nodes that see each others intermittently. Delay tolerant networking not only addresses scenarios in which connectivity is disrupted, it also deals with situations where some links could face large propagation delays or high error rates so that a regular end-to-end transport protocol (e.g., TCP, UDP) trying to cross that link or disrupted area would fail.

Research in DTNs has been inspired by work which took place in Inter-Planetary Networking (IPN) within the IPNSIG working group (IPN Interest Group) created in 1998 and the IPN project at the the National Aeronautics and Space Administration (NASA). Networks being on different planets face intermittent connectivity. Researchers also realized that similar problems could be found in wireless sensor networks. The Delay Tolerant Network Research Group (DTNRG) was then created at the Internet Research Task Force (IRTF) in 2002, and is now leading and federating most of the current work on the topic. Also of interest, the Defense Advanced Research Projects Agency (DARPA) launched a DTN program in 2004, which has supported work in DTNRG.

In the delay tolerant networking scenario that we focus on in this work, nodes are mobile and have wireless networking capabilities. They are able to communicate with each other only when they are in transmission range. The network suffers from frequent connectivity disruptions, making the topology only intermittently and partially connected. Due to these disruptions, regular ad hoc networking approaches for routing and transport do not work, and new solutions must be proposed. In particular, we investigate scenarios where network entities are mobile (e.g., mobile phones, PDAs) and are carried by people sharing social relationships (e.g., students from the same university, students in the same program, or people living in the same city).

In this context, as mobility is driven by social factors, regularities in the interactions between DTN entities exist. Understanding these interactions, i.e., the way the network is connected and disconnected, is of great help for the design of efficient routing protocols. As a consequence, we first present in this thesis work in which we characterise network connectivity in a real life scenario. We highlight that there is heterogeneity in interactions between participants and we show that routing can be improved using this information. Afterwards, we introduce too algorithms which use knowledge about network connectivity to route data from one point to another: the first one uses mean inter-contact times, the second one mobility patterns of nodes. We analyse the performance of these algorithms by replaying real connectivity data in simulations. Finally, as new connectivity data sets are still needed by the research community to better understand the properties of DTN scenarios and to perform protocol evaluations, we present results from an experiment we conducted in which we collected connectivity data that we used in an urban setting.

In this section, we present in detail the context for delay tolerant networking, the research issues and our main contributions to the research domain.

1.1 Context

In this section, we describe the context in which the emerging domain of delay tolerant networking arises. We first review the assumptions of the Internet architecture and protocols. Then, we present the new environments, also called *challenged networks*, in which current communication solutions fail, and for which DTNs are suggested as a solution.

1.1.1 Internet assumptions

The Internet as we use it now has been designed in the context where entities (end-hosts and routers) are mostly connected through wired links. Protocols and applications that were developed thirty years ago are still in use today. They rely on assumptions that we describe here:

- *End-to-end connectivity*: they assume that between any two nodes that can communicate with each other there is a continuous, bidirectional end-to-end path.
- *Short round trip times*: the assumed end-to-end connectivity, mainly relying upon wired links, leads to round trip times ranging from few milliseconds to a second.
- *Symmetric data rates*: data rates are assumed to be roughly symmetric. Even if access networks such as ADSL are asymmetric by nature, the level of asymmetry is suitable with regard to user needs. The up-link, which has the lowest capacity, has been sufficiently provisioned for most of the applications.
- *Low error rates*: transmission errors can occur for diverse reasons such as link failure or congestion, but are considered to be unusual and they arise at a low rate.

1.1.2 Challenged environments

The area of delay tolerant networking addresses scenarios that differ drastically from the Internet in a number ways that might arise individually or collectively:

- *Intermittent connectivity*: Connectivity may suffer from disruptions leading to link failure and network partitioning, for a large number of reasons:
  - *Mobility issues*: Nodes are mobile and can communicate from one to another in a wireless fashion only when within radio range.
  - *Radio issues*: Radio conditions might not be sufficiently good or could be disturbed by external interference.
  - *Battery issues*: Nodes could run out of battery power or use policies that make them unable to communicate for a certain amount of time.
- *Delay issues*: Links could have a very high propagation delay or have such a highly variable delay that traditional protocols like TCP would fail.
- *Asymmetric data rates*: Links can suffer from highly asymmetric data rates or can be simply just unidirectional. This could happened when using satellite links or Data Mules [1].
- *High error rates*: Some links may have high error rates. They could require a high level of correction and a large number of retransmissions, leading to the creation of tight bottlenecks.

With respect to all the constraints just listed, DTNs are often called *challenged networks*. These kind of networks might include:

- *MANETs (Mobile Ad hoc Networks)*: An ad hoc network [2] is composed of several mobile nodes sharing one or several wireless channels without centralized control or an established infrastructure. Each node can communicate directly with the others that are within its transmission range. Each node may act as a router in order to allow communication between hosts that are not within radio range of each other. A typical situation where ad hoc networks are useful is the deployment of a communication system in cases where setting up infrastructure is a non-trivial task or may take too much time. MANETs can be used by public safety forces in a natural disaster, by military forces on a battlefield, or simply by people at a meeting or a conference. These networks suffer from connectivity disruptions due to node mobility and radio propagation issues. Delay tolerant approaches are envisioned to help provide continual communication services in such an unfavorable context.
- *PSNs (Pocket Switched Networks)*: Introduced by the Haggle project [3], Pocket Switched Networks [4] are self-organizing peer-to-peer networks that could take advantage of opportunistic contacts between people. In PSNs, people can relay information for the others

in a store and forward fashion. The Haggle architecture is described in more detail in Sec. 2.1.3. Some people refer to this kind of networks as *contact networks*.

- *Transportation and Vehicular Networks*: These networks are often named VANETs (Vehicular Ad Hoc Networks). They focus on wireless vehicle-to-vehicle and vehicle-to-infrastructure communications. One common scenario is that vehicles can act as store and forward message switches with short-range communication capabilities. Vehicles can exchange information with other cars or with access points from time to time opportunistically. The time between two connections may be long and transfer times could be short because of the high mobility of vehicles.

- *WSNs (Wireless Sensor Networks)*: In this kind of challenged network, nodes have extremely limited power, memory, and CPU capabilities. Communication is often scheduled to conserve power. As a consequence, specific protocols are developed to deal with connectivity disruptions and resource issues.

- *UWSNs (Underwater Sensor Networks)*: These networks consist of a number of devices deployed underwater to perform collaborative tasks in a given area. They could be enabled in oceanographic data collection, pollution monitoring, disaster prevention, assisted navigation or tactical surveillance. In UWSNs, communication takes place through wireless underwater acoustic networking technologies. As a consequence, bandwidth is severely limited, long propagation delays and high errors rates are encountered, and devices could fail at any time for a large number of reasons.

- *Satellite and Inter-Planetary Networks* [5, 6]: As Fall states: “Satellite networking plots very long-distance radio links (e.g., deep space RF communications with light propagation These systems may be subject to high latencies with predictable interruption (e.g., due to planetary dynamics or the passing of a scheduled ship), may suffer outage due to environmental conditions (e.g., weather), or may provide a predictably available store-and-forward network service that is only occasionally available (e.g., low-earth orbiting satellites that pass by one or more times each day).”. In addition, the aim of Inter-Planetary Networking (IPN) is to define the architecture and protocols necessary to permit inter-operation of the Internet resident on Earth with other remotely located Internets resident on other planets or spacecraft in transit. While the Earth’s Internet is basically a “network of connected networks”, the Interplanetary Internet may therefore be thought of as a “network of disconnected Internets”. Inter-networking in this environment will require the development of new techniques [7].

In these contexts, a large portion of the communication protocols that have been designed for the Internet fail. Indeed, TCP and UDP, which are the support of most of the other protocols (POP, HTTP,...) and applications, assume the existence of good quality end-to-end paths between hosts. TCP, for instance, is a conversational protocol because of the three way handshake required to establish a connection, acknowledgements during data transfer and the four way handshake needed to close a connection. The chatty nature of TCP requires a sufficient level of interactivity between the two end-hosts that we may not have in a DTN. In a DTN, TCP fails for two possible reasons: (1) because the probability that an end-to-end path exists at a given time between two nodes that are not in communication range is too low, (2) because propagation delays are so long that TCP considers that it has to stop transmitting to counter a supposed network congestion.

In this work, we mainly address networks that suffer from connectivity disruptions, and in which regularities can be found in the connectivity patterns between nodes. This could be the case in transportation networks such as city-wide bus networks, in pocket switched networks where people have different kinds of interactions depending on their social relationships, or in mobile ad hoc networks used for instance by public safety forces when proximity of people is influenced by their position in the functional organization and hierarchy.

1.1.3 Open issues

Extending the Internet architecture with delay tolerant networking suggests that one or several new protocol stacks have to be developed and that interoperability issues have to be resolved. In this work, we focus on scenarios in which network connectivity is intermittent. Our reference scenario is close to the one addressed in pocket switched networking: devices carried by people on a campus, in an urban setting, or at a conference, are used to exchange data in a peer-to-peer fashion. In this context, a number of networking issues need to be solved:

- *Transport*: As we will see in the next chapter, it is widely accepted that nodes need to communicate in a *store and forward* fashion and that they need to have storage capabilities to carry data on behalf of the others. But a large number of open questions remain in this area. For instance, what kinds of acknowledgement, flow control and congestion avoidance mechanisms should we use?

- *Resource management*: As devices can have limited capacity in terms of batteries, computation power and storage, policies and strategies have to be defined for admission control and buffer management.

- *Addressing*: Mobility of nodes, their possible attachment to a large number of existing networks and the use of different communication interfaces make addressing very challenging. Solutions have to be found to provide an addressing scheme which is resilient in the face of these connectivity issues.

- *Routing*: Being in an environment where nodes see each other intermittently, a complete path may never exist physically between a source and a destination. Routing is then challenging. Routing strategies that take into account the properties of such an environment are thus needed to ensure efficient delivery of messages.
- *Forwarding*: The number of factors that might be integrated in forwarding decisions is large. These factors cover the kind of communication interfaces to use, the willingness of nodes to forward messages for the others, or routing information. Adaptive algorithms have to be developed to take into account all these factors.
- *Interoperability*: DTN enabled nodes may need to pass through Internet-linked networks and interact with legacy applications. Transport protocols and application proxies have to be provided to support this.

In this work, we concentrate on DTN routing by trying to understand properties of network connectivity that could serve routing protocols to make efficient routing or forwarding decisions. We analyse real contact patterns of nodes to answer fundamental questions like: What does connectivity between nodes look like? Do nodes see every node in the same manner or are there differences in interactions that can be characterized? To which point do social relationships influence contact patterns? Is there any simple model that could be used to model these interactions so that it could help the design of efficient routing algorithms?

We have also performed empirical work, in which we propose routing solutions that are based on knowledge of nodes contacts and mobility patterns. The questions we address are the following: Is it possible to achieve message delivery in very large intermittently connected networks? To what extend? What is the trade-off between the energy invested in message transfer and routing performance?

1.2 Contributions

In this work, we present contributions in the domain of delay tolerant networking that share the common property of dealing with the fact that interactions between entities are heterogeneous and display regularities that can be used for routing information. We highlight the presence of heterogeneity in node interactions with the analysis of a real data set and we show that one has an interest in advantage of such heterogeneity. Finally, in a contribution to the on-going effort of trace collection, we present results of an experiment that we conducted to collect connectivity data. The goal of this experiment was to study a content distribution scenario in an urban setting using short range access points.

In more detail, the contributions of this work are the following:

- *The impact of pairwise inter-contact patterns on routing in delay tolerant networks* [8]: Prior work on the understanding of contact patterns in delay tolerant networks has typically focused on inter-contact time patterns in the aggregate. In this work, we argue that pairwise inter-contact patterns are a more refined and efficient tool for characterizing DTNs. First, we provide a detailed statistical analysis of pairwise inter-contact times in three reference DTN data sets. We find that the empirical distributions tend to be well fit by log-normal curves, with exponential curves also fitting a significant portion of the distributions. Second, we investigate the relationship between pairwise and aggregate inter-contact times. In particular, we show how the aggregation of exponential pairwise inter-contacts may lead to aggregate inter-contacts with power laws of various degrees. Finally, we propose a novel single copy opportunistic routing scheme that fully exploits pairwise inter-contact heterogeneity. This algorithm provides the minimum delivery time in case of exponential pairwise inter-contacts. In this scheme, nodes choose to relay messages to neighbors that are closer (in terms of total expected delivery time) to the destination. The scheme is derived analytically in the case of heterogeneous independent exponential inter-contacts and is evaluated on the three reference data sets that we used.

- *DTN Routing in a Mobility Pattern Space* [9, 10]: Routing in a delay tolerant network benefits considerably if one can take advantage of knowledge concerning node mobility. This work addresses this problem with a generic algorithm based on the use of a high-dimensional Euclidean space, that we call MobySpace, constructed upon nodes' mobility patterns. We provide here an analysis and a large scale evaluation of this routing scheme in the context of ambient networking by replaying real mobility traces. The specific MobySpace evaluated is based on the frequency of visits of nodes to each possible location. We show that routing based on MobySpace can achieve good performance compared to that of a number of standard algorithms, especially for nodes that are present in the network a large portion of the time. We determine that the degree of homogeneity of node mobility patterns has a high impact on routing. And finally, we study the ability of nodes to learn their own mobility patterns.

- *Content distribution in an urban setting* [11]: This work investigates the feasibility of a city-wide content distribution architecture composed of short range wireless access points. We look at how a target group of intermittently and partially connected mobile nodes can improve the diffusion of information within the group by leveraging fixed and mobile nodes that are exterior to the group. The fixed nodes are data sources, and the external mobile nodes are data relays, and we examine the trade off between the use of each in order to obtain high satisfaction within the target group, which consists of data sinks. We conducted an experiment in Cambridge, UK, to gather mobility traces that we used for the study of this content distribution architecture. In this scenario, the simple fact that members of the target group collaborate leads to a delivery ratio of 90%. In addition, the use of external mobile nodes to relay the information slightly increases the delivery ratio while significantly decreasing the delay.

1.3 Outline

This thesis is structured as follows. First, Chapter 2 proposes a state of the art in delay tolerant networking by giving an overview of architectures, applications and protocols that have been studied or deployed. Then, Chapter 3, Chapter 4 and Chapter 5 present our contributions in the order they were introduced in Sec. 1.2. Finally, Chapter 6 concludes this work by summarizing the contributions and discussing the future directions in which this work can be extended.

Chapter 2

State of the art

THIS chapter provides an overview of research work and industrial activities related to delay tolerant networking. We first present architectures that have been proposed to support communications when connectivity suffers from disruptions. Then, we present an overview of DTN routing protocols and of real connectivity data sets or mobility models that can be used to evaluate them. Finally, we review applications and real deployments where delay tolerant networking has been or is being applied.

2.1 Architectures

In this section, we present the three main architectures related to delay tolerant networking. We will see that they share common properties such as a *store and forward* manner of delivering information and the packaging of application data into *bundles* or *Application Data Units* (ADUs).

2.1.1 DTNRG Reference Architecture

The DTN reference architecture [12] is a generalization of the IPN (Inter-Planetary Networks) architecture supported by the National Aeronautics and Space Administration (NASA) which is described by Akyildiz et al. [7]. It is being developed by the Delay Tolerant Network Research Group (DTNRG) [13] which is sponsored by DARPA as an architecture [14, 15] to support messaging in DTNs. The architecture consists mainly of an overlay, called the *bundle layer*, added above the transport layer in the classic TCP/IP stack. Messages, or ADUs, in the DTNRG architecture are called *bundles*. They are transferred atomically between DTN nodes in a hop-by-hop fashion using a transport protocol that ensures node-to-node reliability. A node is said to get the *bundle custody* when it receives a bundle. Bundles can be of any size and nodes are assumed to have large buffers in which they can store them.

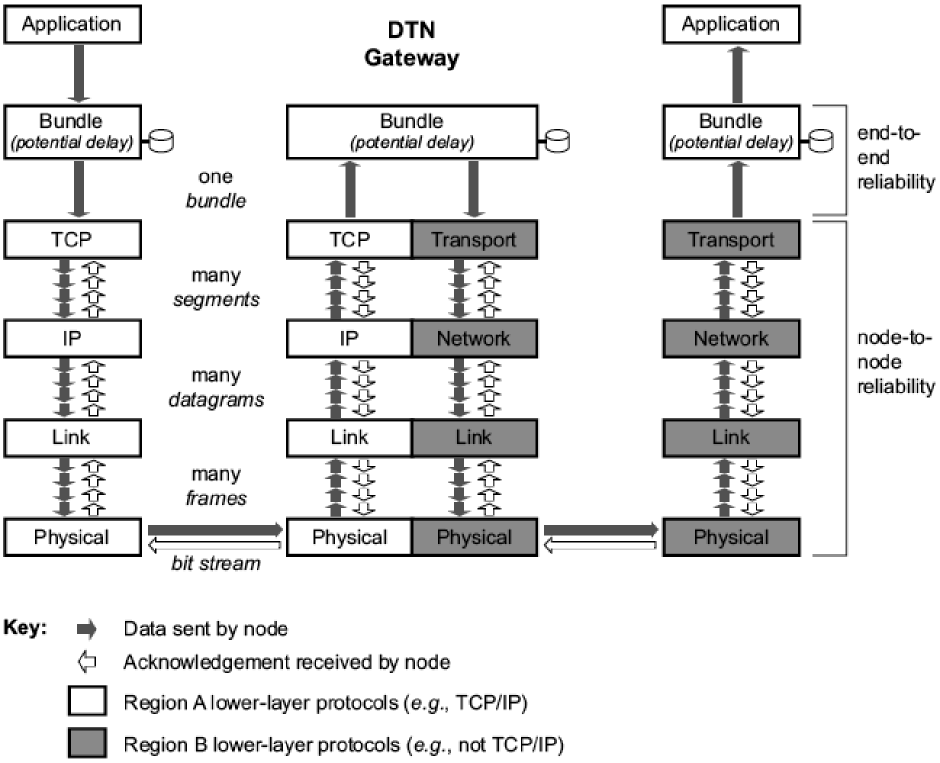


Figure 2.1: DTNRG reference protocol stack [16].

Fig. 2.1 shows the protocol stack defined in the DTNRG architecture. Application data are packaged by applications into bundles that are transmitted by the bundle layer in a hop-by-hop fashion from one DTN gateway to another. The network is divided into networking regions bounded by DTN gateways that ensure bundle's passage through the regions. An example of such organisation is presented by Fig. 2.2. Within each region, a different protocol stack can be used. A bundle can, for instance, be carried to the destination having been transported using TCP over a Wi-Fi link, using a reliable transport protocol such as SCTP [17] over a satellite link, and, finally, using a basic transport protocol over avian carriers [18] providing reliable data acknowledgements. In this transmission chain, large delays could be introduced for instance if there is a need to wait for the satellite to be in communication range or simply because of an unsuccessful transmission attempt with the avian carrier (length of time-outs triggered when waiting for an acknowledgment might go from few minutes to days depending on the environment).

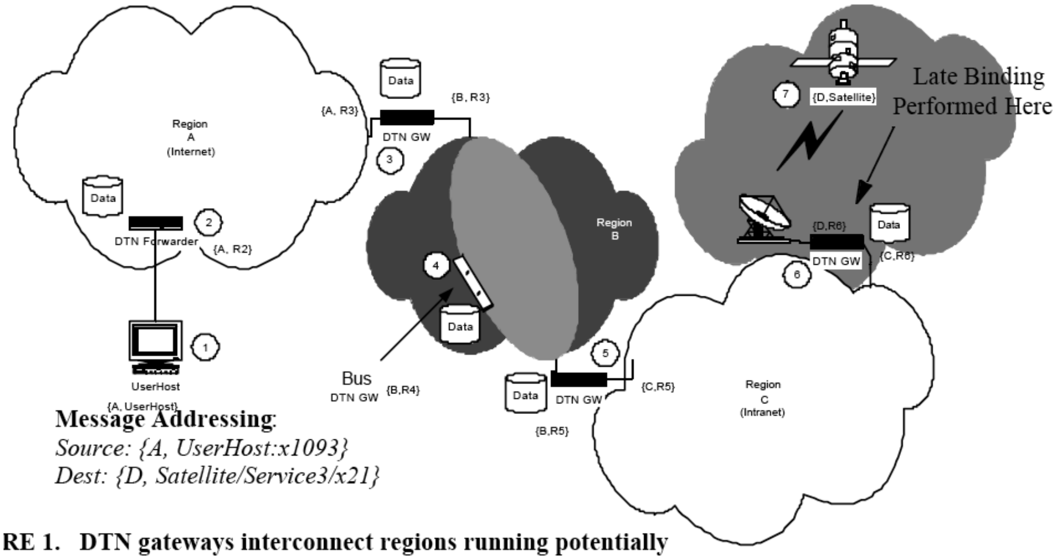


Figure 2.2: Regions interconnected by DTN gateways [5]

To route messages, *name tuples* are defined to describe where end nodes are. These tuples are composed of the region name and the object (end node) name. Each gateway is able to locate objects within the regions it is taking care of using a DNS-like approach. Taking Fall's example [5], an example tuple would be, in the case of the Internet:

```
{internet.icann.int, http://www.ietf.org/oview.html}
```

The DTN architecture provides for postal-like services. Priorities can be attributed to bundles such as *bulk*, *normal*, and *expedited*. Delivery options are also available such as, for instance, *Report When Bundle Received*, which asks for a report to be sent when any DTN nodes receives the bundle, *Report When Bundle Forwarded*, which asks intermediate nodes to report that they have forwarded the bundle, or *Report When Bundle Delivered*, which asks the destination to report to the source upon reception of a bundle.

2.1.2 Tetherless Communication Architecture

Seth et al. [19] recently proposed an architecture for *tetherless communications*, which provides support for opportunistic communications that are resilient to disconnections and aware of mobility. The Tetherless Communication Architecture (TCA) extends the DTNRG architecture in two ways. First, it translates the notion of session state between sources and destinations to networks where connectivity is disrupted by providing session persistence across disconnections. Second, it supports node mobility by providing the means to locate nodes in order to make them reachable at all times.

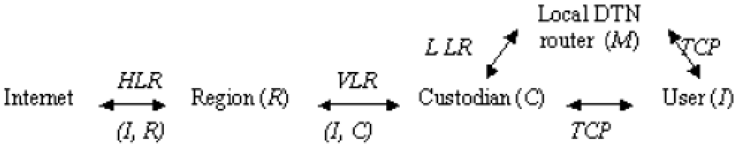


Figure 2.3: Tetherless Communication Architecture [19]

TCA also uses regions to demarcate connectivity islands and introduces mechanisms to locate mobile devices and to perform routing. Each mobile node possesses a Globally Unique Identifier (GUID) so that the node can be addressed from anywhere. Registries, organized in a hierarchical manner, are used to map GUIDs to the current locations of nodes. Fig. 2.3 presents this structure. Note that the Internet is defined as a specific region. A DHT (Distributed Hash Table) is maintained in this region by a set of Home Location Registries (HLR), to map GUIDs (I) to node locations (R). Each region R maintains one or more Visitor Location Registers (VLRs) which map each GUID to a *custodian node* C, to which end hosts are attached. Custodian nodes are routers that are always available and that can store data on behalf of disconnected mobile nodes. A mobile user can be directly connected to a custodian router or can pass through another level, which is a Local DTN router. A Local DTN router, which can be a Custodian node as well, is usually mobile and ensures data transportation from one point to another (e.g., a bus). Custodian nodes maintain a Local Location Register (LLR) to map GUIs to local routers. The registries are updated when nodes move, which allows devices to be reached and information to be routed between mobile nodes.

The Opportunistic Connection Management Protocol (OCMP) [20] proposed by Seth et al. is used to provide persistent session state on top of DTNs. This is inspired by the Persistent Connectivity Management Protocol (PCMP) proposed by Ott et al. [21]. It allows applications to opportunistically communicate on multiple network interfaces, switch across interfaces, aggregate their bandwidth, remain disconnected or powered off for arbitrarily long periods, pack-

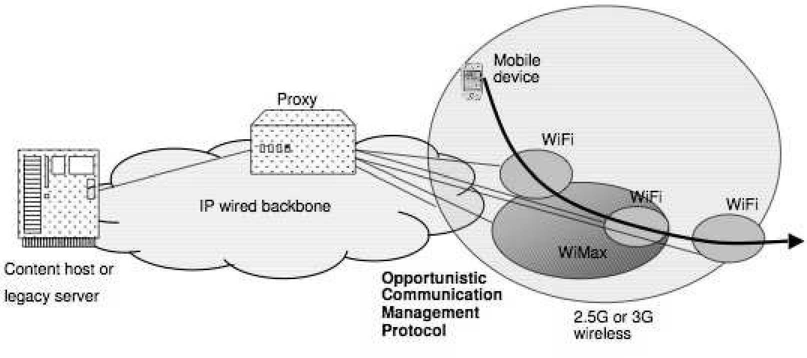


Figure 2.4: Opportunistic Connection Management Protocol (OCMP) [22]

age and move their state across devices, and inter-operate with legacy applications and servers. The implementation is in Java so that it can be run on any Java-based mobile device.

Legacy protocol support is plugin based. Support for HTTP and FTP have been provided. Higher level operations such as blogging and picture publishing have been developed using the APIs provided by Blogger[1] and Flickr[2]

2.1.3 Pocket Switched Networking Architecture

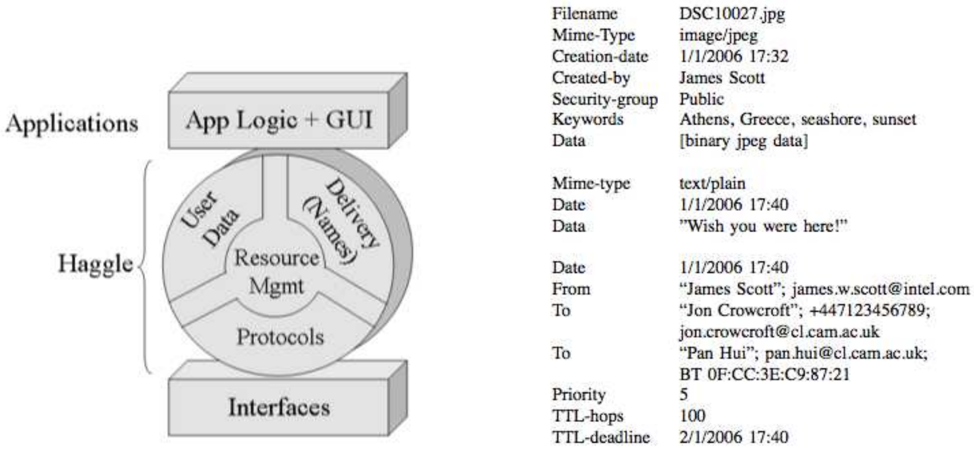
The European IST Haggle project [3] addresses also issues related to intermittent connectivity by targeting the three methods by which data can be transferred between devices that people carry every day: through an existing infrastructure, in an ad hoc fashion with neighboring nodes, and using the mobility of users to carry the data. They refer to the scenario they study as Pocket Switched Networking (PSN) inspired by the idea that people carry packets in their pocket. The Haggle infrastructure is able to opportunistically take advantage of transfer opportunities and to manage the limited resources of nodes.

Haggle proposes to forward ADUs (Application Data Units, i.e., bundles in the DTNRG architecture) using application layer information (e.g., email addresses, names, etc.) rather than a network-specific identifier. As in the DTNRG architecture, Haggle adopts store-and-forward operations to handle asynchronous communications and is able to use intermediate nodes to relay application data.

Fig. 2.5(a) presents an overview of the architecture. As one can see, nodes have storage capabilities to store user data received from applications and potentially from other nodes. Nodes have a transverse module for resource management and a forwarding engine that uses

---

[1] `http://www.blogger.com`
[2] `http://www.flickr.com`



Filename	DSC10027.jpg
Mime-Type	image/jpeg
Creation-date	1/1/2006 17:32
Created-by	James Scott
Security-group	Public
Keywords	Athens, Greece, seashore, sunset
Data	[binary jpeg data]
Mime-type	text/plain
Date	1/1/2006 17:40
Data	"Wish you were here!"
Date	1/1/2006 17:40
From	"James Scott"; james.w.scott@intel.com
To	"Jon Crowcroft"; +447123456789; jon.crowcroft@cl.cam.ac.uk
To	"Pan Hui"; pan.hui@cl.cam.ac.uk; BT 0F:CC:3E:C9:87:21
Priority	5
TTL-hops	100
TTL-deadline	2/1/2006 17:40

(a) Haggle Architecture (b) Application Data Unit

Figure 2.5: Haggle propositions [3].

user-level names. Finally, a communication module takes care of communications by hiding connectivity issues from the application. Fig. 2.5(b) shows an example of an ADU. We can see that Jon Crowcroft, being the receiver, is addressed using his professional contact data. This ADU is an email containing an image and some text.

2.2 Routing protocols

As adding delay tolerance to networks impacts every level of protocol stacks and almost all node services, listing all past and on-going work related to DTNs would take too much space. We focus here on routing propositions, as they are strongly related to the contributions we present in this work.

As mentioned by Jain et al. [23], routing is one of the very challenging open issues in DTNs. Because the network suffers from connectivity issues, MANET [2] routing algorithms such as OLSR, based on the proactive broadcast of control information, or AODV, which uses *on-demand* broadcasted route requests, fail to achieve routing. Different approaches have to be found.

Following the taxonomy introduced by Jones et al. [24] we divide routing propositions for DTNs into three main categories: *replication* based, *knowledge* based and *hybrid* strategies. Replication based approaches take advantage of what we can call *node diversity*. They address ways information can be disseminated among several carriers to increase the chance that it would reach the destination. Knowledge based strategies use of information that nodes obtain about connectivity or network conditions to make efficient forwarding decisions that improve routing performance. Hybrid approaches, as suggested, use both replication and knowledge.

DTN routing strategies face a general trade-off between different factors: robustness to connectivity disruptions (i.e., the delivery ratio), the impact on network resources (i.e., the routing overhead), and the rapidity of execution (i.e., the message delay). One might want a DTN routing process to be as robust as possible, to have the lowest impact on the network and to be as fast as possible. From our experience, as soon as we try to improve on all these performance metrics, we must consider the nature of the underlying network connectivity. We borrow terminology the DTNRG architecture to describe different patterns of node contacts:

- *Scheduled contacts*: these can exist, for instance, between a base station somewhere on earth and a low earth orbiting relay satellite.
- *Predicted contacts*: these are not scheduled, but predictions of their existence can be made by analyzing past information or using hypotheses regarding node movements.
- *Opportunistic contacts*: these are created simply by the presence of two entities at the same place, in a meeting that was neither scheduled nor predicted.

2.2.1 Replication based

Replication based routing protocols do not assume that nodes have knowledge that could be used to make forwarding decisions.

Epidemic routing, referred sometimes to *flooding*, proposed Vahdat and Becker [25], is one of the most basic and popular solutions when nothing is known about the behavior of nodes. In this scheme, when any two nodes meet, they compare bundles and each sends the other the ones it lacks. The ZebraNet project [26] makes use of epidemic routing in its studies of animal migration and inter-species interactions. Data are flooded in the network so that they reach access points as a result of animals' mobility. Epidemic routing is very high in resource consumption, but it achieves the best performance in terms of delivery ratio and delay. As, in large networks, it may be too costly in terms of energy consumption or memory usage, we discuss work that attempt to limit flooding or to use coding to make better use of replication.

Grossglauser et al. [27, 28] were among the first to introduce a replication based scheme for MANETs that is not epidemic. They show that the total network throughput can be increased using mobility of nodes if applications tolerate some delay for data delivery. Packets can be buffered at nodes and they can be routed using one intermediate relay or transfered directly to the destination.

To control flooding in DTNs, Spyropoulos et al. have introduced the Spray and Wait [29] protocol that distributes a number of copies to relays and then waits until one of them meets the destination. Harras et al. [30] have evaluated simple controlled message flooding schemes with heuristics based, for instance, on hop limits or timeouts. They also introduce a mechanism based on packet erasure. Once a message arrives at the destination after basic flooding, the remaining copies in the buffers of other nodes are erased.

Wang et al. [31] propose to reincode messages with erasure codes and distribute their different parts over a large number of relays, so that the original messages can be reconstituted even if not all packets are received. Widmer et al. [32] have explored network coding techniques. All these approaches distribute multiple copies of packets and ensure a high reliability of delivery, and a low latency, but they imply high buffer occupancy and high bandwidth consumption.

2.2.2 Knowledge based

Since relaying information can lead to buffer overflows and inefficient use of transmission media or transmission opportunities, one would prefer to use information that can be obtained about node interactions to help make less naive forwarding decisions.

Some work has been done with scheduled contacts, such as the scheme described by Jain et al. [23], which tries to improve the connectivity to the Internet of an isolated village based on knowledge of when a low-earth orbiting relay satellite and a motorbike might be available to make the necessary connections. They introduced a spectrum of schemes that use more or less knowledge from connectivity and from the network. This starts with the First Contact (FC) scheme in which nodes have no knowledge, and where, at each hop, bundles are forwarded to random neighbors. Other algorithms make use of more and more knowledge of the network conditions and connectivity. MED (Minimum Expected Delay) is able to assign to node pairs, the cost being the sum of the average waiting time, propagation delay and transmission delay. In MED, Bundles are source routed using Dijkstra's algorithm. Then, a linear programming based algorithm makes use of all information about connectivity and present and future traffic demands. They show through simulations that, as predicted, the more knowledge nodes have, the better is the routing performance. Also of interest is work by Akyildiz et al. [7] on interplanetary networking, where they use scheduled contacts, such as the ones between planets, in the framework of a DTN architecture.

Work has been performed with predicted contacts, such as the algorithm of Lindgren et al. [33], which relies on nodes having a community-based mobility pattern. Nodes mainly remain inside their community and sometimes visit other communities. To route a bundle to a destination, a node can transfer that bundle to a node that belongs to the same community as the destination. This work has been extended to define ProPhet [34], a protocol that is being tested in Sweden within the Sámi Network Connectivity project [35] and is being standardized in the IRTF DTN group. In a similar manner, Burns et al. [36] proposed a routing algorithm that uses past frequencies of contacts. Also making use of past contacts, Davis et al. [37] improved the basic epidemic scheme with the introduction of adaptive dropping policies. Recently, Musolesi et al. [38] have introduced a generic method that uses Kalman filters to combine and evaluate the multiple dimensions of the context in order to take routing decisions. The context is made up of measurements that nodes perform periodically, which can be related to connectivity patterns, but not necessarily. This mechanism allows network architects to define their own hierarchy among the different context attributes.

LeBrun et al. [39] propose a routing algorithm for vehicular DTNs using current position and trajectories of nodes to predict their future distance to the destination. They replay GPS data collected from actual buses in the San Francisco MUNI System, through the NextBus project. Finally, Jones et al. [40] propose a link state routing protocol for DTNs that uses the Minimum Estimated Expected Delay (MEED) as the metric.

Some research projects, such as Data Mules [1], SeNTD [41] or Message Ferrying [42], use mobile network elements to transport data from fixed sensors to a number of access points in an opportunistic fashion. For instance, in SeNTD, data from sensors placed on buoys that monitor the water quality on a lake are relayed by tourist tour boats or pleasure cruisers. Also of interest, Zhao et al. [43] propose to calculate routes for ferries such that the traffic demands are met and the data delivery delay is minimized.

2.2.3 Hybrid approaches

All the knowledge based approaches presented above are single-copy. This means that messages can be routed in a multi-hop fashion in the network but, at every time, only one copy of the message is present in the network. To be robust to node failures and to increase routing performance, we might want some replication. Hybrid solutions are therefor desirable.

Jain et al. [44] have tried to determine an optimal allocation of erasure codes blocks over multiple paths to maximize the probability of delivery depending of path failure probabilities and level of redundancy. Given the path success probabilities, code blocks are spread over the multiple possible paths. They applied two different algorithms: one which finds the k best edge-disjoint paths and one which aims at maximizing a specific objective function using a linear programming method.

Note that, in addition to the single-copy protocol that we introduce in this work which is based on the use of mobility patterns of nodes, we present an extended version, which is a controlled flooding solution that performs knowledge based forwarding decisions. This work is presented in Chapter 4.

2.3 Mobility issues

To evaluate a DTN scheme, one has three possibilities: (1) implement and deploy the mechanism, which could cost a lot and could be limited in scale, (2) perform simulations with mobility models, which could be too synthetic and thus unrealistic, and (3) replay, in simulation, mobility traces gathered in real contexts, which could lead to too scenario-specific evaluations. None of these methods is ideal. However, the increasing number of data collection efforts and the advances in mobility modelling are leading to higher quality frameworks for DTN scheme evaluation. This section presents a short overview of these recent efforts.

2.3.1 Data sets

We present here a non exhaustive list of data sets available to the research community. Most data sets that could be of interest to the DTN community are now available and centralized by the CRAWDAD [45] (Community Resource for Archiving Wireless Data At Dartmouth) project at Dartmouth University.

Dealing with traces allows researchers to better understand the underlying properties of DTN environments. In the contributions presented in this work, we performed all the evaluations in simulations with real mobility traces and we emphasize the understanding of these traces with detailed statistics.

2.3.1.1 GPS based

GPS is at first sight the natural mean to collect mobility data. However it does not work when a user is indoors. As a consequence, it has only been used to monitor outdoor activities such as the taxi cabs from the MPT Radio Taxi company in Warsaw, Poland. Sarafijanovic-Djukic et al. [46] obtained GPS coordinates for 825 taxis over 92 days in an area of $60 * 48$ km.

2.3.1.2 Bluetooth based

Experiments using Bluetooth contact loggers, namely Intel motes (iMotes), have been conducted within the Haggle [47] project, which explores networking possibilities for mobile users using peer-to-peer connectivity in addition to existing infrastructures. They deployed these iMotes on different sets of people (conference attendees, corporate employees, groups of friends) these iMotes to measure and characterize interactions (i.e., the timing of contacts) between people. As this thesis also presents a study based on a similar iMote experiment that we conducted, we provide a detailed comparison of the different existing data sets in Chapter 5.

2.3.1.3 Wi-Fi based

Work has been conducted around Wi-Fi access networks to gather data that can be used, after some processing, as DTN-like data. Dartmouth College [48] has deployed one of the most extensive trace collection efforts to gather information on a Wi-Fi access network. These data have been used as mobility data to characterise the mobility of users [49, 50] or to evaluate DTN routing protocols [10]. Other Wi-Fi access networks have been used to analyse mobility such as that of ETH Zürich [51] and that of the Wireless Topology Discovery project at UCSD [52]. The usual hypothesis that one makes to infer interactions between Wi-Fi nodes is that two nodes are in contact whenever they are attached to the same AP (Access Point). We use such data throughput this thesis and we discuss the reasonableness of the simplifying assumptions in Chapters 3, 4 and **??**.

Additionally, in the UMass DieselNet project [53], which aims to study DTN routing in a transportation setting, a testbed to gather interactions between 40 buses in western Massachusetts was deployed in 2005. Buses are equipped with 802.11b interfaces working in ad hoc mode.

2.3.1.4 GSM based

The Reality Mining [54] experiment, conducted at MIT, has captured proximity, location, and activity information from 100 subjects over an academic year. Each participant had an application running on their mobile phone to record proximity with others through periodic Bluetooth scans as well as their location, using the identifier of the cell to which the phone is attached.

2.3.1.5 Social activity based

Some researchers have inferred theoretical interactions from information about human behavior. For instance, Srinivasan et al. [55] have studied interactions between students based on their university time tables.

2.3.2 Mobility models

Due to the limited number of data sets available and the fact that they are generally specific to a scenario, synthetic models have been widely used. Models such as Random Walk (i.e., Brownian motion [56]) or Random Way-Point [57] have been very popular. However, when compared to real mobility data, the properties generated by models often do not match. As a consequence, researchers are proposing synthetic models that intend to reproduce statistical distributions or behaviors that have been observed in reality, as discussed by Borrel et al. [58].

For instance, Musolesi et al. [59] propose a model in which movements of nodes are driven by the social relationships among them. Bohacek [60] designed a mobility model of individuals in urban settings based on a recent US Department of Bureau of Labor Statistics time-use study. Legendre et al. [61] question whether microscopic mobility behaviors are valuable to represent mobility with more realness and their influence on important characteristics (e.g., link duration distribution).

2.4 Applications

Regarding applications that use opportunistic communications, the large majority of them have been deployed for feasibility studies, while a few are currently working to provide operational services. We first present operational deployments and continue with the prospective ones.

2.4.1 Operational deployments

In this section, we present some applications (commercial and non-profit) that use DTN communications and that are deployed in running operational systems. These deployments concern urban peer-to-peer messaging systems and wildlife monitoring systems.

2.4.1.1 Urban peer-to-peer messaging

BUZZeeBee [62] is a commercial messaging service for wireless devices and desktops using Wi-Fi, Bluetooth, and Ethernet. When one person using BUZZeeBee encounters another, a spontaneous wireless connection is automatically created. Messages can be addressed to a specific user or target a group of users. They are relayed by other wireless devices. Every communication is anonymous unless the sender has included his business card in the message. Each message exists by default during only 24 hours but a user can pay to increase the time-to-live of his messages. This is to avoid having useless messages pollute the network and being spread far from the place in which they might be useful. Since it is a commercial product, stores can send messages to users including coupons and special offers.

MobiLuck [63] is similar to BUZZeeBee but focuses more on the fact that it can help people to meet each other. Currently, users can send messages for free that are relayed by the nearby Bluetooth devices. They can send their profile to other MobiLuckers, including their photo. Soon, MobiLuck will allow for multi-user games, and the downloading of ring tones, logos, music and games. They plan to encourage the use of such applications in professional exhibitions, to help people exchange business cards and find new customers or jobs. MobiLuck claims one million users in 200 countries. To counter the low user density issues, they define

MobiLuck HotSpots which are merely meeting places for MobiLuckers such as a restaurant, a theater, a nightclub, a store, etc.

ProxyDating [64] is a dating service available on Bluetooth enabled mobile phones. Users configure their profile. The application detects matching persons in the proximity and triggers. The phone rings and the face of the other appears on screen. These functionalities are often mentioned as matchmaking services.

Socialight [65] allows people to leave virtual sticky notes anywhere in the world and share them with others. A virtual sticky note is a piece of content that is associated with a geographic position that could be text, pictures, videos or sound clips. The visibility of virtual sticky notes can be restricted to a single user, to group of users or to everybody. The application is available on mobile phones and PDAs. People can thus leave their impressions of the museum they just visited or advise others that there is a good restaurant in the neighborhood. Since Socialight is a company, advertisement nodes could be let for commercial purpose.

The growing number of smart phones with Wi-Fi or Bluetooth will support the popularity of these kinds of applications. People predict that they will be very popular among groups of teenagers or in countries where communication between people men and women is very regulated. However, as shown by the following message, found on a discussion forum[3], the low density of users is currently a barrier to the success of that kind of applications:

```
>I really want to use BUZZeeBee. It’s a cool new app using Wi-Fi, a mix of
>social/dating networking, classifieds and optin advertising for one’s favorite stores.
>But it’s also like IM, you need other people to use it with because it’s not online,
>but offline peer to peer and proximity based!
>So if you can give it a try...let me know. I’m in Mountain View, California.
>Thanks

If you supply the equipment needed, furnish us with return plane tickets and overnight
accommodation, I’m sure many of us here in the U.K. would gladly spend time with you over a coffee.
:ˆO:ˆO:ˆO
```

2.4.1.2 Monitoring

ZebraNet [26] is a project that aims at monitoring zebras in Africa. It was one of the first deployments where researchers posed the problem of collecting data as a networking problem. The idea emerged in 2002 and the deployment was done in 2004 at Mpala [4] which is a biology field station in central Kenya that Princeton University administers along with the Kenya Wildlife Service. Before this project started, people in wildlife monitoring had three possibilities:

---

[3] `http://forums.vnunet.com/thread.jspa?messageID=612828`

[4] `http://www.nasm.edu/ceps/mpala/main.html`

- *GPS trackers*: Collars equipped with a GPS tracker are attached to animals and record GPS positions. The problem is that collars have to be collected and redeployed periodically to collect data. It may disturb animals behaviors and create gaps in the data.
- *Satellite tracking*: Collars that the animals wear can be equipped with an active satellite tracking system. This demands high battery capacity and offers a very low bit rate to report collected information. Batteries have to be changed periodically, which complicates the deployment as for GPS trackers.
- *VHF*: Transmitting collars using VHF can be used to determine the locations of animals using triangulation methods. The precision of this positioning system is very dependant on the radio propagation in the area the animals traverse.

ZebraNet uses low-powered devices that communicate in a peer-to-peer fashion and have data storage. GPS positions are sampled periodically and stored locally with other information such as sun/shade indication, activity (standing or moving), speed, step rate, etc. Information is then distributing among zebras and dropped opportunistically at collecting devices that researchers place close to where the zebras live. This system allows then to collect a large amount of different kinds of data and to reduce the maintenance and collection activities. Note that the ZebraNet scenario is similar to wireless sensor networks in which data needs to be collected at sinks.

The Delay Tolerant Shell (DTS) [66], proposed by Lukac et al., has been tested in an experimental setting within the Middle America Subduction Experiment (MASE) where 50 seismic stations are deployed and connected with long range Wi-Fi links. DTS is a shell that can be used for node administration and data collection. It has the particularity of being tolerant to connectivity disruptions by relying on the fact that data are stored and then forwarded in an epidemic fashion. In the environment where the experiment takes place, the end-to-end path availability is evaluated by authors to be just 81%.

2.4.2 Prospective deployments

Here, we present some of the applications that have been launched or that are intended to be launched in support of academic research on DTNs.

2.4.2.1 Transportations

The UMass *DieselNet* [67] project has deployed hardware in buses of the town of Amherst, Massachusetts, with the aim of providing services to users and the bus company. They are currently studying opportunities of transfer that a possible network architecture could benefit from. Buses upload information to a central server whenever they are in range of any open 802.11 AP. The information includes bus-to-bus transfers, GPS logs, and the downloading of software updates. 60 days of traces are available to the community. For each transfer opportunity, they recorded the duration, transferred data, location, bus speed and direction. They recently improved the network connectivity with *throw-boxes* which consist of long range (thus low bitrate) ad hoc nodes that are able to communicate with an infrastructure network and with buses. The ultimate aim of this experiment is to provide internet services and experiment with new kinds of applications.

Lebrun et al. [68] propose to use what they call BlueSpots to distribute content in buses on the campus at UC Davis. They are currently working on the testbed.

The CafNet (Carry-and-Forward Networking) system is being developed in the CarTel [69] project. It aims at developing and deploying a protocol stack similar to the DTN reference implementation in a context where data mules have high capacity and experience long periods of disconnection. CarTel uses data mules to transfer large amounts of sensor information from and to mobile nodes. They mainly target applications in automobiles, such as road traffic monitoring, on-board automotive diagnostics and notification, road surface diagnostics and opportunistic data transport in general.

2.4.2.2 Urban settings

Karlsson et al. [70] deployed a proof-of-concept demonstrator for the concept of delay tolerant broadcasting. Public channels are available to send and receive content opportunistically. Nodes such as mobile phones or laptops are constantly looking for other devices using Bluetooth or Wi-Fi. Nodes can potentially relay content that they themselves are not interested in. They implemented various strategies for soliciting content to see how to improve the performance of the system.

The RollerNet [71] project that we conducted at LIP6 with Farid Benbadis in parallel to this thesis work aims to analyse and use mobility in rollerblade tours in Paris, to offer services that will be available to participants. Every Friday evening and every Sunday afternoon in Paris, weather permitting, groups of between 5,000 and 15,000 people go rollerblading for three hours at a time. They are guided by staff members and assisted by public safety forces. In order to analyze the mobility of participants, we perform experiments in which we deploy iMotes, the Bluetooth contact loggers (thanks to James Scott from Intel Research Cambridge), on approximately a hundred volunteers, these could be organisers' friends, members of rollerblading associations or members of staff. In addition to this sensor deployment, we ask other people to activate Bluetooth on their mobile phones. The iMotes log, at a high frequency, the devices (other iMotes or people's mobile phones) that they meet. The data that we collect allows us to measure and characterize the interactions between people over the duration of the rollerblade tours. Such information is helpful in the design of new forms of applications in the domains of emergency response, location services, and content delivery. A preliminary proof-of-concept of a positioning system has already been demonstrated [72].

2.4.2.3 Monitoring

The Diverse Outdoor Mobile Environment [73] project at UMass seeks to advance internet technology to deploy in environments where providing networking to mobile users is a challenge. DieselNet was one of the projects conducted within this framework. Two other recently launched projects are:

- *TurtleNet*: This project aims at monitoring the habitat of wood turtles. The primary goal is to collect GPS information, temperature, sunlight intensity, and turtle going on dates. Turtles will be equipped with 6 MHz Mica-2 Dot motes.
- *Underwater monitoring*: In collaboration with oceanographers, the project will conduct underwater monitoring of coastal areas, including sea life and ocean bottoms.

2.4.2.4 Developing regions

The Sámi Network Connectivity project [35] aims to establish Internet communications for the Sámi population of reindeer herders, who live in remote areas in the north of Sweden. The initial goal is to provide email, cached web access, reindeer herd tracking telemetry and basic file and data transfer services. Bundles are routed between DTN gateways, which could be mobile or fixed. Fixed gateways may be in residential communities where the internet can be accessed. The proposed solution is DTN-based, and is currently being developed by the Internet Research Task Force DTN Research Group (IRTF-DTNRG). New routing algorithms such as ProPhet [33] and protocols are being developed to take advantage of the forwarding opportunities presented by periodic chance encounters.

Deployments of rural kiosks [74] using the DTN reference architecture is being experimented by Seth et al. within India to provide services to remote population at low costs. Kiosks will provide a variety of services such as birth, marriage, and death certificates, land records, and consulting on medical and agricultural problems. The infrastructure will support opportunistic and scheduled links established with buses that pass through the remote villages and internet-connected areas.

The aim of the TIER [75] (Technology and Infrastructure for Emerging Regions) project at the University of California at Berkeley is to address the challenges in bringing the information technology revolution to the masses in developing regions. [76]

Chapter 3

The impact of pairwise inter-contact patterns on routing in delay tolerant networks

THIS chapter presents our first contribution. The main argument in this work is that researchers should also be looking at heterogeneous inter-contact time distributions when modeling DTNs and designing routing algorithms.

3.1 Introduction

In delay tolerant networks (DTNs) [12] nodes are typically mobile and have wireless networking capabilities. They are able to communicate with each other only when they are within transmission range. The network suffers from frequent connectivity disruptions, making the topology only intermittently and partially connected. This means that there is no guarantee

---

This is joint work with Vania Conan (Thales Communications) and Timur Friedman (Université Pierre et Marie Curie, LiP6–CNRS).

that an end-to-end path exists between a given pair of nodes at a given time. Examples from the recent literature include the DieselNet project [67], which features communication devices deployed in a regional bus system, and Pocket Switched Networks (PSNs) [50], which are formed by devices that people carry every day, such as cell phones, PDAs, and music players. In contexts such as these, end-to-end paths can exist temporarily, or may sometimes never exist, with only partial paths emerging.

Routing in such a context is a challenge, and much depends upon what one expects in terms of node mobility. Initial DTN work focused on exploiting scheduled meeting times [23]. Focus then turned to the sort of randomness in meeting times encountered in mobile ad-hoc networks [30, 77], and characterised in mobility models such as Random Way-Point [57], and Random Walk [56]. These models yield homogeneous patterns, where all nodes share a single inter-contact time distribution. More recent work has analysed experimental data sets [48, 50, 54] that record actual inter-contact patterns in a number of different environments. Chaintreau et al. [50] have observed power laws of various degrees, including degrees lower than 2, where the mean and/or variance are infinite. They analyse so-called Spray and Wait protocols, modelling pairwise inter-contact times with power laws, and conclude that such schemes do not allow finite time delivery of messages when the degree is lower than 1. At the same time, routing schemes that have been tested through simulation on the same data [10] have proved effective. This constitutes what we call the *power law paradox*.

In this work, we advocate that researchers look at pairwise inter-contact patterns. We make three contributions along these lines: First, we provide a detailed statistical analysis of pairwise inter-contact patterns in three reference DTN data sets. Previous work has studied inter-contact times in the aggregate, across all pairs of nodes. It has combined, and thus obscured, the individual effects of pairwise inter-contacts. We characterize heterogenities in contact times and inter-contact times, and find that distributions of inter-contact times tend to be well modeled by log-normal curves. Exponential curves also tend to fit a fair portion of distributions.

Second, we provide an explanation for the DTN power law paradox. We describe how distributions with finite means and variances can be composed to yield the heavy-tailed distributions that Chaintreau et al. [50] observed. In particular, we show that exponential pairwise inter-contact time distributions can be combined to create an aggregate power law distribution.

Third, we propose a single copy opportunistic routing scheme that fully exploits pairwise inter-contact heterogeneity. This algorithm provides minimum delivery time in case of exponentially distributed pairwise inter-contact times. As we shall see, the heterogeneity that we highlight allows us to usefully extend the work of Spyropoulos et al. [78, 79], which analyzes numerous routing schemes for DTNs, but that uses mobility models that yield homogeneous distributions. The scheme provides an opportunistic version of the *minimum expected delay*

(MED) routing introduced by Jain et al. [23], whereby a node relays a message to a neighbor that is closer, in terms of total expected delivery time, to the destination. We formally analyse the scheme for the case of heterogeneous independent exponential inter-contacts, we evaluate it through simulation on the three reference data sets.

The rest of this work is structured as follows. Sec. 3.2 provides a statistical analysis of pairwise contacts in the three real life data sets we used in this work. Sec. 3.3 presents the power law paradox. Sec. 3.4 introduces our novel opportunistic routing scheme, and Sec. 3.5 provides its evaluation. Sec. 3.6 describes related work concerning routing and mobility in DTNs. Sec. 3.7 concludes the chapter, discussing directions for future work.

3.2 Pairwise interactions

This section introduces and analyses the three different data sets that we use in the rest of this work. We characterize interactions that may occur in DTN scenarios and highlight the different kinds of heterogenities that arise.

3.2.1 Experimental data sets

We describe here the contexts in which the data sets have been collected and the acquisition methodologies that were used. All of these data sets are publicly available in the CRAWDAD archive [45].

3.2.1.1 Dartmouth data

This connectivity data set has been inferred from traces collected in the Wi-Fi access network of Dartmouth College [48]. The traces that we use were pre-processed by Song et al. for their prior work [80] on mobility prediction. They track users' sessions in the wireless network, noting the time at which nodes associate and dissociate from access points. Although the Dartmouth data is not from a DTN network, we use it because it is perhaps the richest data set publicly available that tracks users in a campus setting, and because of its quality. Jones et al. [40], Leguay et al. [10], and Chaintreau et al. [50] have recently used these traces in a similar way.

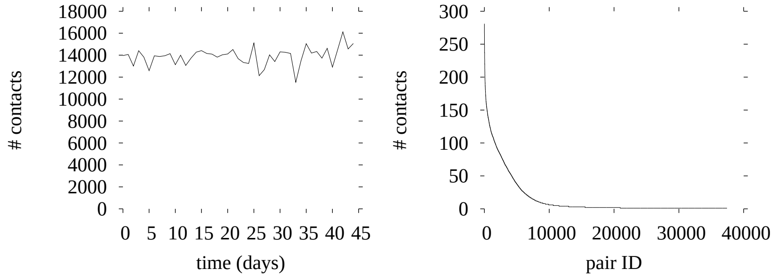
A few judicious assumptions are required to adapt the Dartmouth data for DTN studies. First, we only consider the subset of users who were present in the network every day between January 26th 2004 and March 11th 2004, an academic period during which we expect nodes' activity to be fairly stationary. This data set contains 834 users, or nodes. Then, we assume that two nodes are in contact if they are attached at the same time to the same access point (AP). We miss other contacts between users that are not logged by Wi-Fi devices, because the users are not carrying the devices or have turned them off. These contacts might have been logged in a true DTN network, by lighter-weight wearable devices that remain on at all times. When more extensive DTN data in campus settings becomes available, researchers will need to revisit the studies made using the Dartmouth data, to see if the lack of such contacts has an impact on their conclusions. Finally, we filter the data to remove the well known *ping-pong* effect. Wireless nodes, even non-mobile ones, can oscillate at a high frequency between two APs. To counter this, we filter all the inter-contact times below 1,800 seconds (30 minutes). Note that defining better filtering methods, although challenging, would be of interest for the community. As this is not the purpose of this work, we choose here the threshold that Yoon et al. [81] used for the same purpose. We use this inferred data set for the remainder of this chapter.

Fig. 3.1 presents, for all the data sets, the evolution over time of the total number of contacts that occurred between nodes (left column) and the number of contacts for every pair of nodes having at least one contact, ranked in decreasing order (right column). Fig. 3.1(a) and Fig. 3.1(b) are the plots for the Dartmouth data set. As Fig. 3.1(a) shows, the interactions between nodes are quite stable over time. We observed 13,901.7 contacts per day on average, with a standard deviation of 796.9 contacts. We conjecture that this stability comes from the fact that we choose only nodes that are present every day. Fig. 3.1(b) shows that a few node pairs had a high number of contacts, and that this number then decreases very rapidly. Just 10.7% (i.e., 37,424) of node pairs had contacts between each other, and these are the ones that are plotted. Among these, the mean number of contacts was 15.4, with a standard deviation of 32.9 contacts.

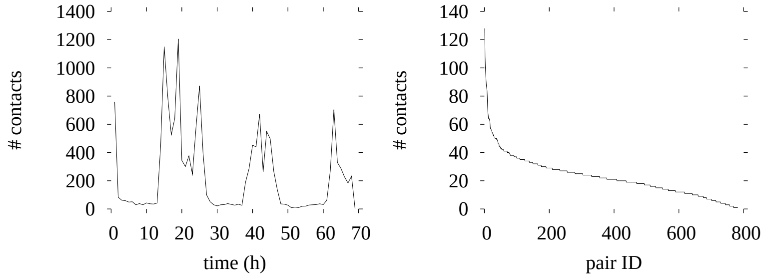
3.2.1.2 iMote data

Chaintreau et al. [50] used iMotes (Bluetooth contact loggers from Intel) to acquire proximity contacts that occurred between participants in the student workshop at the *Infocom 2005* research conference. Students were asked to carry one of these sensors in their pocket at all times. Due to Bluetooth's short range, authors logged instances when people were close to each other (typically within 10 meters). They collected data from 41 iMotes over 3 days. The devices performed Bluetooth inquiry scans every 2 minutes. For each pair of nodes (i,j), we considered that i and j were in contact if either one saw the other. Note that, as with the Dartmouth data, many contacts might be missed. Those that occur between the 2 minute scans are not registered, and two nodes that are scanning simultaneously will not see each other.

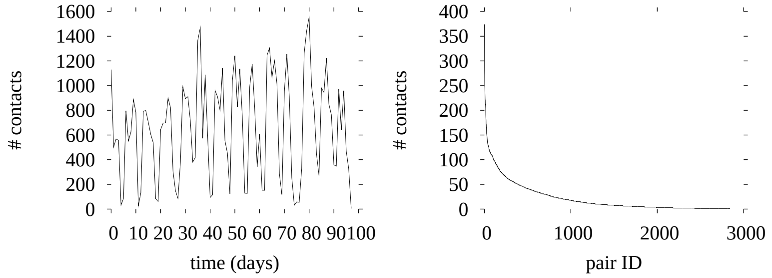
In this data set, the evolution of the number of contacts between participants shows diurnal variations, as seen in Fig. 3.1(c). We observed 231.7 contacts per hour on average, with a standard deviation of 281.3 contacts. Fig. 3.1(d) only plots node pairs that had contacts, but



(a) Dartmouth (b) Dartmouth



(c) iMote (d) iMote



(e) MIT (f) MIT

Figure 3.1: Evolution of the total number of contacts over time (left). Number of contacts for each pair of nodes (right); pairs are sorted in decreasing order of their number of contacts.

these represent fully 95.4% of the pairs. For these pairs, the mean number of contacts was 22.8, with a standard deviation of 14.8 contacts. The iMote data shows more contacts for the typical node pair than does the Dartmouth data.

3.2.1.3 MIT data

The Reality Mining experiment [54] conducted at MIT captured proximity, location, and activity information from 97 subjects (mainly students) over the course of an academic year. Each participant had an application running on their mobile phone to record proximity with others through periodic Bluetooth scans (every 5 minutes) in a similar fashion to that of the iMote experiment. Locality information comes from knowing which GSM network cell the phone is attached to. We only make use of the Bluetooth proximity data to determine whether two nodes were in contact. We selected 95 days of data corresponding to the first semester of the academic year 2004-2005 where activity was high in the traces in terms of the number of phones that collected data and the number of contacts that were recorded.

Fig. 3.1(e) displays weekly variations in the number of contacts between participants. The mean number of contacts per day is 660.0 contacts per day, with a standard deviation of 405.072 contacts. The number of interactions is lower than in the iMote data set, where the mean number of contacts per day was 1,378.39, and that was among only 41 nodes. For the 60.4% of node pairs that had at least one contact, and are plotted in Fig. 3.1(f), there is a mean of 22.3 contacts, with a standard deviation of 32.8 contacts. This plot more closely resembles the plot for the Dartmouth data set than the iMotes data set.

These data sets represent three different DTN scenarios which are of interest for the understanding of interactions between people that might carry communication devices. We will refer to these data sets as *Dartmouth*, *iMote* and *MIT*.

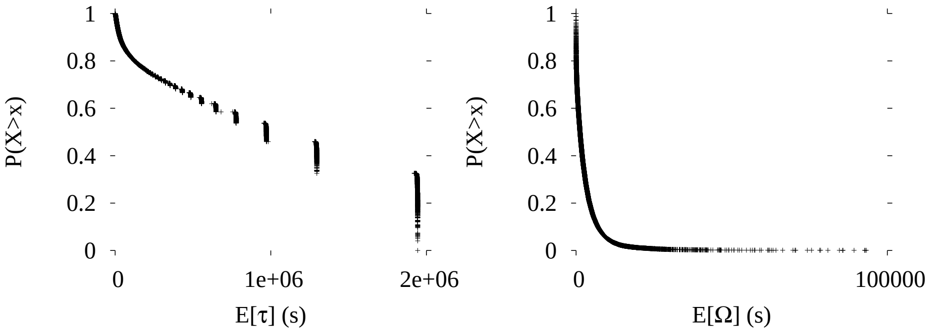
3.2.2 Heterogeneity in expectations

This section looks at the durations of contacts between pairs of nodes (*contact times*) and the time that elapses between any such contacts (*inter-contact times*). We focus on heterogeneity, looking at the distributions for all node pairs. In the data sets just described in Sec. 3.2.1, we have already observed heterogeneity in the number of contacts per node pair. However, a deeper look is required to understand the impact of contact patterns on routing.

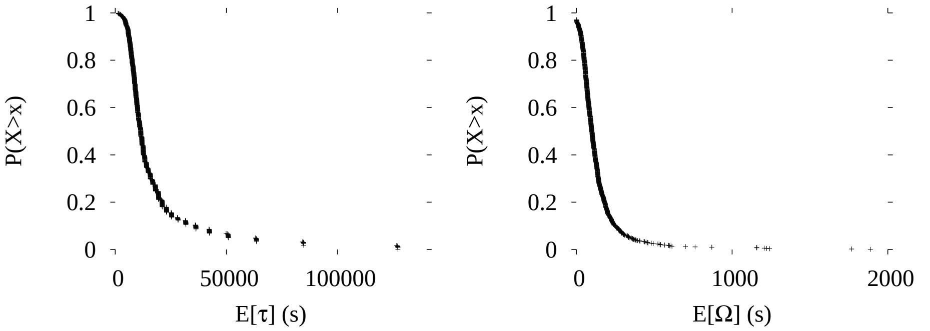
Fig. 3.2 shows, in the left column, the cumulative distribution, for all node pairs, of mean inter-contact times. We denote with $\mathrm{E}(\tau)$ the expectation of inter-contact times, with τ being the process of inter-contact times for a given pair. Similarly, Fig. 3.2 shows the distribution of $\mathrm{E}(\Omega)$, the expected contact times of node pairs. We can see that the distributions are heterogeneous, with the means spanning over three orders of magnitude. The mean inter-contact time is 280.6 hours for Dartmouth, with a standard deviation of 210.5 hours; 4.9 hours for iMote, with a standard deviation of 5.6 hours; and 387.1 hours for MIT, with a standard deviation of 377.3 hours.

The mean contact times are also heterogeneous, as shown by plots in the right column of Fig. 3.2. The mean expected contact times are: 0.8 hours for Dartmouth, with a standard deviation of 3.0 hours; 0.03 hours for iMote, with a standard deviation of 0.04 hours; and 0.3 hours for MIT, with a standard deviation of 0.4 hours.

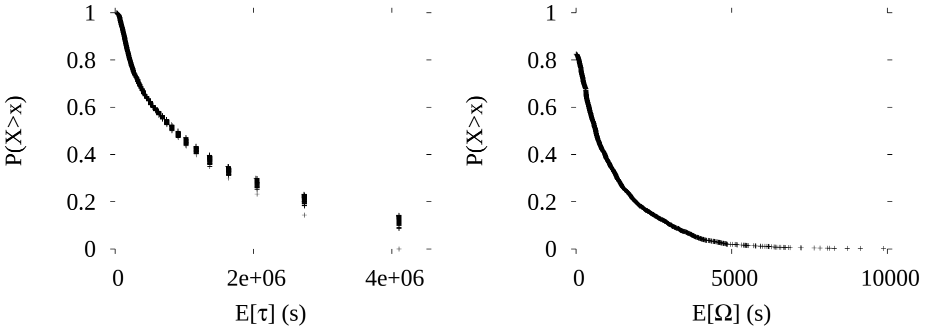
We also observe, for all three data sets, that mean contact times are much shorter than inter-contact times. This leads us to conjecture that understanding the inter-contact times processes is more crucial that understanding the contact-times processes if one needs to choose which to focus on. We can already see that the iMote context seems more suitable for applications



(a) E(τ) on Dartmouth (b) E(Ω) on Dartmouth



(c) E(τ) on iMote (d) E(Ω) on iMote



(e) E(τ) on MIT (f) E(Ω) on MIT

Figure 3.2: CDF of mean inter-contact times E(τ) (left) and mean contact times E(Ω) (right).

that would perform direct transfers between source and destination nodes, as pairwise average inter-contact times are lower than in the other data sets. Such a simple scheme might not work in MIT and Dartmouth, where average inter-contact times are very large. Routing mechanisms might have to be proposed. We address the issue of routing in more detail in Secs. 3.4 and 3.5.

3.2.3 Nature of inter-contact times distributions

To better understand the inter-contact time processes between node pairs, we test for whether the distribution of inter-contact times between any two nodes can be modelled either by an exponential, a log-normal, or a power law (to be precise, Pareto) distribution.

For this purpose, we use the Cramer-Smirnov-Von-Mises [82] statistical hypothesis test. Recall that such a statistical test can only *reject* or *fail to reject* a given hypothesis. So, when the hypothetical distribution is rejected by the test, we are certain that the distribution computed over the data does not match. In the other hand, when the test fails to reject the hypothesis, we only know that this is true to a confidence level $1 - \alpha$. We used a relatively high level of confidence ($\alpha = 0.01$) and also visually cross-checked the goodness of fits.

For each pair of nodes (i,j) having at least 4 contacts, we compare the cumulative distribution I_N^{ij} of the N inter-contact times observed and the hypothesis functions whose cumulative distributions are given by the three following formulas:

- Exponential distribution: $F_{ij}(x) = 1 - e^{-\lambda_{ij}x}$
- Pareto distribution: $F_{ij}(x) = 1 - \left(\frac{x_{\mathrm{m}ij}}{x}\right)^{k_{ij}}$
- Log-normal distribution: $F_{ij}(x) = \frac{1}{2} + \frac{1}{2}\mathrm{erf}\left[\frac{\ln(x) - \mu_{ij}}{\sigma_{ij}\sqrt{2}}\right]$

Note that, for a given node pair, several distributions may fit the inter-contact distribution. We see an example in Fig. 3.3 of the inter-contact times for an iMote node pair. These inter-contact times are found to follow a log-normal distribution.

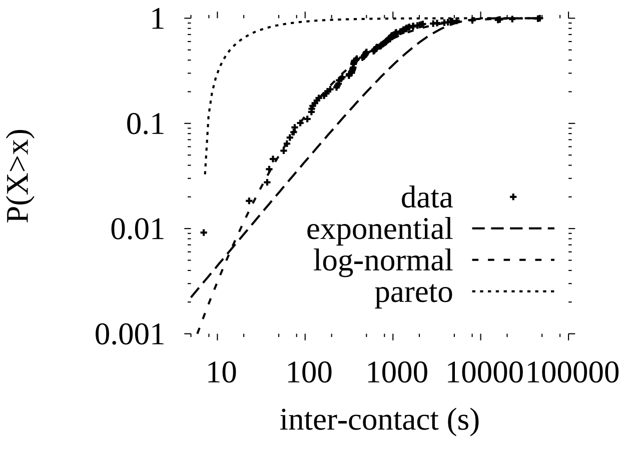


Figure 3.3: Fitting results for a given node pair in iMote data.

Table 3.1 presents, for each data set, the proportion of pairs for which the distribution of inter-contact times fits an exponential, a Pareto, and a log-normal distribution. We also show the proportion of pairs that were rejected for all three hypothetical distributions.

One notable observation is that log-normal tends to fit better than exponential or Pareto for all three data sets. The main reason is that the log-normal distribution offers a more versatile model to capture the variability in inter-contact patterns across the different pairs of nodes. Almost no pair of nodes has been found fit only an exponential or a Pareto distribution. For Dartmouth, for example, 0.1% of node pairs are exponential only, and the same proportion are Pareto only, while 36.4% of node pairs only match a log-normal distribution.

	Dartmouth	**iMote**	**MIT**
Number of pairs tested	20,211	755	2,174
Exponential	42.8 %	7.9 %	56.3 %
Pareto	34.2 %	12.3 %	26.5 %
Log-normal	85.8 %	99.4 %	96.9 %
None	12.9 %	0.4 %	2.7 %

Table 3.1: Fitting results.

Fig. 3.4 plots the distribution of the σ parameters of the log-normal in all data sets. This parameter governs the shape of the cumulative distribution at the origin. σ values are higher than 1, which means that asymptotes tend to be vertical at the origin. σ values are higher in Dartmouth, with an average of 3.5, compared to 2.2 for iMote and 2.1 for MIT. Considering the fact that the log-normal distribution can cover a large spectrum of asymptotic behaviors at the origin, subsets of log-normal distributions could well be considered to be exponentially or power law distributed.

From these observations, it seems reasonable, in these data sets, to consider pairwise inter-contact time distributions as log-normal rather than power law or exponential. This speaks to the heterogeneity of the distributions. The log-normal family is better capable of modeling the variations of behaviors across the pairs of nodes. The reasons are probably twofold. First, it covers a large span of asymptotic behaviors at the origin (from horizontal to vertical asymptotes). Second, it can capture light tailed behavior as well as some heavy tailed behavior, while always maintaining a finite expectation and/or variance (contrary to power laws with degrees lower than 1).

As we have examined only three data sets, albeit often-used ones, we cannot draw firm general conclusions about what will be revealed elsewhere. But one might reasonably expect that other mobility traces captured in similar environments will show similar characteristics.

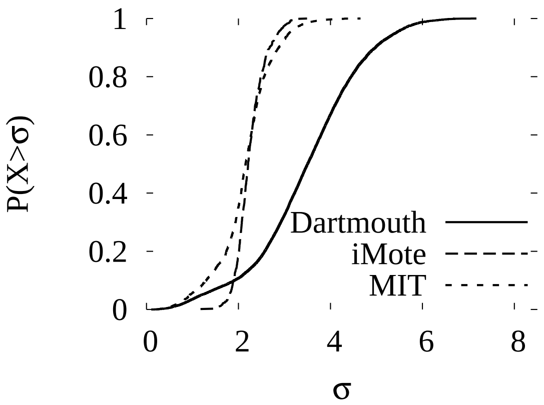


Figure 3.4: Distribution of σ_{ij} for log-normal node pairs in data.

3.3 The power law paradox

Chaintreau et al. [50] report that aggregated inter-contact times follow power laws in a number of DTN traces (including ones based on Dartmouth and iMote data). At the same time, we have just ruled out a power-law distribution to model pairwise inter-contacts. And this holds for traces based on the same Dartmouth and iMote data. This constitutes what we called the *power law paradox*.

Computing the cumulative distribution of aggregated inter-contact times for the Dartmouth data set confirms this observation. The plot in Fig. 3.5 shows that it follows a power law of the form $f(x) = cx^{\delta}$, with exponent $\delta = -0.16$ and scale parameter $c = 3.45$.

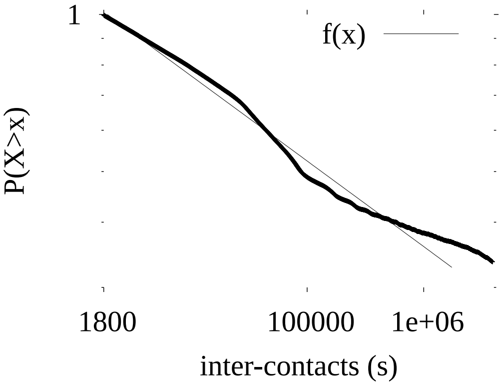


Figure 3.5: Distribution of inter-contacts.

This section studies this power law paradox by looking at the relationship between pairwise inter-contacts and inter-contacts aggregated over the entire set of pairs of nodes. We show how specific combinations of heterogeneous pairwise inter-contact times can explain the aggregated power laws that have often been observed in experimental data sets.

First, we introduce a generic model of a heterogeneous DTN. The model supports any inter-contact time distribution. To analyze the power law paradox, we study the special case where the distributions are exponentials. This leads to explicit analytical formulas that provide insight into the phenomenon at play. The exponential hypothesis is not the best fit for the data we analysed, but appears often as the second best choice (as one can see from Table 3.1). It constitutes also the extreme case – exponential decay being the very prototype of light tailed distributions – and the opposite of the heavy tailed behavior of the power laws. We are then able to formally derive the aggregate distribution of inter-contact times in the case of exponential pairwise inter-contacts. Confronting the formula with the experimental Dartmouth data set, we confirm that power-laws in the aggregate are compatible with exponential pairwise inter-contacts.

3.3.1 Heterogeneous DTN Model

Let us first consider a generic model for a heterogeneous DTN composed of *n* nodes. All nodes are given a unique ID in $1, 2, ..., n$. For any two nodes (i,j) we denote as $(t_{ij}^{(n)})_{n \in Z} = ... < t_{ij}^{-1} < t_{ij}^{0} < t_{ij}^{1} < ...$ the sequence of time instants at which a contact between i and j occurs.

The inter-contact pattern of the network is defined in the following way:

- For each pair of nodes (i,j) the pairwise inter-contact sequence $(t_{ij}^{(n)})_{n \in Z}$ is a renewal process; in other words, inter-contact times between nodes i and j are independent identically distributed ***(iid)*** random variables, let's say T_{ij}. Note that for each pair (i,j) the distribution law of T_{ij} and its parameters may differ.
- A given joint distribution of the $n(n-1)/2$ pairwise inter-contact sequences serves in particular to characterize the possible correlations between two inter-contact sequences.

We associate two processes to the model. First we define $T = \inf_{i<j} T_{ij}$ which represents the meeting time between any two nodes in the network; the renewal process this induces gives the "pulse" of inter-contacts. Second we define $\Theta = \cup_{i<j} T_{ij}$ to be the aggregate inter-contact time for all pairs of nodes.

The major hypothesis made by the model is that inter-contact time distributions are stationary. In other words, node behaviors are assumed to change on a slower scale than message exchanges. Nodes are also assumed to have infinite capacity in bandwidth and storage. As we have seen in Fig 3.2, by looking at the number of node contacts, this hypothesis is a drastic simplification for the iMote and MIT data sets, whereas is appears more realistic in the Dartmouth data set.

The model focuses on the temporal dynamics of node connectivity in a DTN. It does not model node mobility directly, but captures inter-contact patterns. In this way it provides a common framework to analyse different DTNs. The traditional Random Way Point and Random Walk mobility models for ad-hoc networks fall in this category (see Carreras et al. [83]). More generally, the model would allow one to capture the different forms of heterogeneity that we have identified in the data sets of Sec. 3.2.

3.3.2 Aggregated power law

In this section we focus on the heterogeneous exponential case, which corresponds to the model with the following complementary hypotheses: the pairwise contact sequences are homogeneous Poisson processes (HPPs), i.e., the inter-contact times follow exponential laws with parameters λ_{ij}. Furthermore the HPPs are independent. The purpose is here to focus on the heterogeneity of pairwise mean inter-contact times, which are given by $1/\lambda_{ij}$ and to study the effect of aggregating the inter-contact patterns. Choosing the exponential case may be seen as an extreme; the tail distribution of the exponential is the very opposite of the heavy tailed pattern that power laws capture best. Moreover, it may seem that the pairwise exponential assumption is too strong to yield a power law in the aggregate.

Let Θ be the aggregate inter-contact time for all pairs of write nodes. Let's write $K = n(n-1)/2$ and renumber the pairwise inter-contacts T_k from 1 to K. We then have $\Theta = \cup_{1 \leq k \leq K} T_k$. Let's imagine that all inter-contact processes are exponentially distributed with various parameters λ_k. Different distributions of the λ_k parameters can model different global properties of DTNs. For example, in some cases, a node will meet most of the others several times a day, and the remaining ones on a weekly basis. Let p_k denote the proportion of pairs with parameter λ_k.

Conditioning on the event that the in the aggregate $\cup_{1 \leq k \leq K} T_k$ the pair is pair number k, we have:

$$P(\Theta > t) = \sum_{k=1}^{k=K} P(T_k > t) p_k \tag{3.1}$$

In the case of a DTN with a very large number of pairs (such as in the Dartmouth case) we can write the same formula with a continuously varying probability distribution $p(\lambda)$ of the λ parameters:

$$P(\Theta > t) = \int_{\lambda=0}^{\infty} e^{-\lambda t} p(\lambda) d\lambda \tag{3.2}$$

What Eq. 3.2 says is that, for the exponential case, the aggregate inter-contact time distribution is fully characterized by the distributions of the λ parameters. More precisely, the tail cumulative distribution of the aggregated inter-contact times is given by the Laplace transform of the distribution p of the λ parameters.

A power law behavior, with parameter α, will appear in the aggregate when the distribution of Θ follows a Pareto law with shape parameter $\alpha > 0$ and scale parameter $b > 0$, in which case we have, for $t \geq 0$:

$$P(\Theta > t) = (\frac{b}{t+b})^{\alpha} \tag{3.3}$$

Since the Laplace transform is invertible, Eq. 3.2 tells us that taking the inverse Laplace transform of $P(\Theta > t)$ gives the distribution p of the λ parameters. We then have, Γ being the Gamma function, for $\lambda \geq 0$:

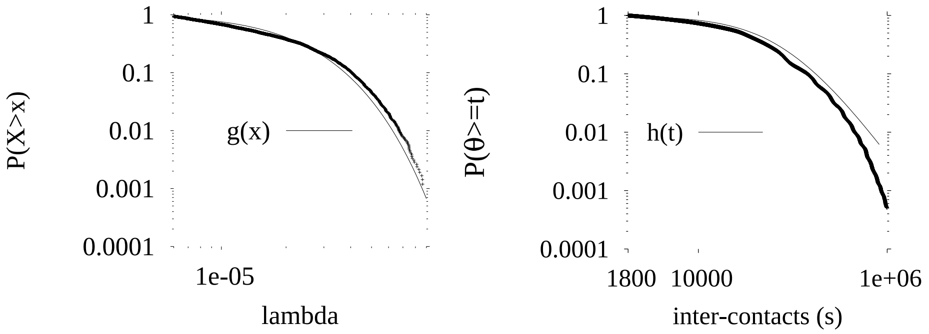
$$p(\lambda) = \frac{\lambda^{\alpha-1} b^{\alpha} e^{-b\lambda}}{\Gamma(\alpha)} \tag{3.4}$$

We can now state the following result: in the exponential case, the aggregate distribution will exhibit a power-law tail cumulative distribution with parameter α if the λ parameters follow a Gamma distribution with the same parameter α.

This answers our initial question: even if all pairwise inter-contacts follow an exponential distribution, it is still possible to regain the power law distribution in the aggregate. One could have thought *a priori* that it would require the distribution of the λ parameters to be a power law, or at least heavy tailed. In that case there would still have been a power law behavior, not directly in the pairwise inter-contacts, but at a global scale of the DTN (its distribution of parameters). In fact Eq. 3.4 shows that this is not necessary, since the tail of the Gamma distribution is asymptotically exponential, and thus Gamma is not heavy tailed.

Let us apply this result to the Dartmouth data set; recall that inter-contact patterns are not all exponential, so to validate the result, we proceed in the following way: we estimate parameters α and b from the cumulative distribution of the λ parameters for pairs that were shown to follow an exponential behavior (the ones that "pass" the Cramer hypothesis test). We find b = 113,766.9 and $\alpha = 2.26$. Fig. 3.6(a) shows the estimated cumulative gamma distribution $g(x)$ with the experimental lambda cumulative distribution for all pairs that have shown to be exponential. Then, we plot in Fig. 3.6(b) the corresponding power-law $h(t)$ with cumulative distribution of aggregated inter-contact times. As one can see, the two experimental curves fit the theoretical curves.

What this result shows is that when one considers an exponential DTN, we can regain the power law behavior for the aggregated inter-contacts when the distribution of the parameters is



(a) λ_{ij} (b) Inter contacts

Figure 3.6: Distributions with exponential pairs.

a Gamma, which is the case in the data we used when considering the subset of pairs that have inter-contact times exponentially distributed.

The important lesson that we can draw is that it is not necessary to introduce pairwise power laws to generate aggregate power laws. This is good news for routing in DTNs. Chaintreau et al. [50] have reported that pairwise power-laws (in particular for degrees lower than 2) have an adverse impact on the opportunistic Spray and Wait routing strategy, casting some doubt on the mere possibility of efficient finite-time delivery of messages across a network. With pairwise inter-contacts with both first moments (mean and variance) finite, the picture looks much brighter, as we shall see in the next section.

3.4 Opportunistic minimum delay

In this section we propose a novel single copy routing strategy that minimizes the delivery delay of messages across the DTN. We formally analyse the scheme for the case of the exponential DTN model introduced in Sec. 3.3.1.

The major hypothesis the model introduces for routing is that contacts (and thus message transfers) are assumed to be instantaneous. The reasonableness of this assumption is supported by the fact that inter-contact times have been observed (see Sec. 3.2) to be much longer on average than contact times. The results with the proposed model are upper bounds, but, as we will see, still provide valuable information and insight on how to route messages in DTNs.

In the routing strategy, message transfers are assumed to be instantaneous. We also consider that nodes know all pairwise mean inter-contact times for all nodes in the network, i.e., each node knows the λ_{ij} matrix. This knowledge could be learned by each node from past contacts and diffused through an epidemic style of routing. The impact of inaccurate or partial knowledge of the interconnectivity matrix is evaluated in Sec. 3.5.

We formally analyse the routing scheme for the exponential DTN case. The results apply in a somewhat more general setting that corresponds to what Carreras et al. [83] designate the *Marks Memoryless* class. The difference with the exponential case is that $T = \inf_{i<j} T_{ij}$, the "pulse" of inter-contacts is not restricted to the exponential case, but can be any distribution law. Note that in this case, all pairwise inter-contact times follow the same law (different from that of the pulse), but with different parameters. The pulse distribution may also introduce correlations between the pairwise inter-contacts.

The proposed routing scheme exploits the opportunistic relaying principle of the Spray and Wait strategy introduced by Grossglauser and Tse [28] in the context of mobile ad-hoc networks. Sec. 3.4.1 derives the formula for expected delivery delay for the one relay single copy Spray and Wait scheme. Sec. 3.4.2 proceeds with the presentation of the opportunistic mean expected delay scheme.

3.4.1 Spray and Wait routing

The Spray and Wait strategy consists of two steps. First, the source node uses the first nodes encountered as relays to the destination. This is the "spraying" step. A relay node then uses the "wait" strategy to relay the message, i.e., it waits until it meets the destination to deliver the message. Here, we study the case where only one relay is used, which we designate 1-SW.

Let us first consider the spraying step. The message is injected at source s at time instant t. The first node r it encounters may be any of the $n-1$ other nodes $d, r_1, r_2, ..., r_{n-2}$ and the time X it takes to meet this first node is the infinum of the inter-contact times with all other nodes:

$$X = inf(R^t_{sd}, R^t_{sr_1}, ..., R^t_{sr_{n-2}}) \tag{3.5}$$

Since all $R^t_{sr_i}$ are independent exponentials with parameters λ_{sr_i}, we have (see [84, p.328]):

- The random index r of the first node encountered is independent of the first encounter time X
- X is exponentially distributed, with parameter: $\Lambda_s = \lambda_{sd} + \sum_{i=1}^{n-2}(\lambda_{sr_i})$
- Pr($First\ node\ encountered\ is\ r$) $= \frac{\lambda_{sr}}{\Lambda_s}$

This means that we can represent the spraying step as independently identifying the encountered node (with probability $\frac{\lambda_{sr}}{\Lambda_s}$) and adding an exponential waiting time with parameter Λ_s. [1]

---

[1] The decoupling of mean waiting time and identity of the encountered node is the key to the derivation of the scheme. In the case of the *Marks Memoryless* class, this result is provided by Wald's Lemma.

Two cases may arise: either the first node encountered r equals d, and s delivers the message with expected time $\frac{1}{\Lambda_s}$, or $r \neq d$ and node r waits to meet node d to deliver the message.

Let's evaluate the time it takes for r to meet d and deliver the message. If the message is received by node r at time t (let's say), its delivery time is equal to R^t_{rd}, the remaining inter-contact time before the next contact between nodes r and d. The memoryless nature of exponentials implies that R^t_{rd} follows an exponential distribution with the same parameter λ_{rd} as the inter-contact time. The mean expected delivery time for a message at node r awaiting delivery to d is thus given by:

$$E[D^w_{rd}] = 1/\lambda_{rd} \tag{3.6}$$

The total delivery time Z_r along path r, i.e., conditioned on using node r as a relay, is thus the sum of the first encounter time X and $E[D^w_{rd}]$ the remaining delivery time between nodes r and d and thus:

$$E[Z_r] = \frac{1}{\Lambda_s} + \frac{1}{\lambda_{rd}} \tag{3.7}$$

The total delivery time Z is computed by conditioning on all possible first encountered nodes $d, r_1, r_2, ..., r_{n-2}$, events whose probabilities are given by $\frac{\lambda_{sr}}{\Lambda_s}$.

After simplification, this leads to the following mean delivery time for 1-SW:

$$E[D^{1-sw}_{sd}] = \frac{(1 + \sum_{r \neq s, r \neq d} \frac{\lambda_{sr}}{\lambda_{rd}})}{\sum_{r \neq s} \lambda_{sr}} \tag{3.8}$$

3.4.2 Minimum delay Spray and Wait

The minimum delay routing strategy is based on the previous 1-SW scheme. First, we derive 1-SW* which sprays the message only to *neighbors* (i.e., nodes encountered) of the source that minimise the expected delivery time in case of pairwise exponential inter-contacts. Second we show that recursively applying the scheme leads to a fixed point that minimises the delay in the case of an arbitrary number of intermediate relay nodes.

Instead of considering all neighbors of source node s as candidate relays, as in the 1-SW scheme, let's consider that the source node s sprays the message only to nodes in a subset R. We call this a 1-SW^R scheme. Following the same line of reasoning as in Sec. 3.4.1, and defining $1/\lambda_{dd} = 0$, one finds that the expected delivery time is given by:

$$E[D^{1-sw^R}_{sd}] = \frac{(1 + \sum_{r \in R} \frac{\lambda_{sr}}{\lambda_{rd}})}{\sum_{r \in R} \lambda_{sr}} \tag{3.9}$$

We define 1-SW* to be a 1-SW^R scheme which uses a subset R that minimizes $E[D^{1-sw^R}_{sd}]$.

Brute force minimization amounts to testing all subsets R of neighbors of source node s. The complexity of the algorithm is exponential in the degree d_s of node s. The structure of Eq. 3.9 allows for the definition of an algorithm which is linear in d_s (see Sec. 3.8.1). To find the subsets R of neighbors of source node s that minimize $E[D_{sd}^{1-sw^R}]$ (Eq. 3.9), we propose the following algorithm:

```
for every destination $d$ do
    Sort its neighbors in increasing mean inter-contact times, in which case we have:
    $0 \leq \frac{1}{\lambda_{1d}} \leq \frac{1}{\lambda_{2d}} \leq \ldots \leq \frac{1}{\lambda_{nd}}$
    Initialise the result set $I = \oslash$ and corresponding minimal mean delivery time (using set $I$)
    $c_I = \frac{1}{\lambda_{1d}}$
    for $i = 1, \ldots, n$ do
        Add node $i$ to set $I$ and compute $E[D_{sd}^{1-sw^I}]$ (as in Eq. 3.9)
        If this value is strictly larger than $c_I$, remove node $i$ from $I$ and stop
        Otherwise, place this value in $c_I$
    end
end
```

At the end, the optimal set of nodes in I and the corresponding minimal delay in c_I. Proof of the algorithm is provided in Sec. 3.8.1.

Let's now introduce a second relay node in the Spray and Wait scheme. Node s now chooses the best first encountered neighbors based on the assumption that they will relay the messages following the 1-SW relay scheme. For a given set of first encountered neighbors R, the total expected delay is given by:

$$E[D_{sd}^{2-sw^R}] = \frac{(1 + \sum_{r \in R} \lambda_{sr} E[D_{rd}^{1-sw}])}{\sum_{r \in R} \lambda_{sr}} \tag{3.10}$$

Minimising Eq. 3.10 for all sets R of neighbors of s is obtained by applying the same algorithm as for 1-SW, since Eq. 3.10 is deduced from Eq. 3.9 by replacing $\frac{1}{\lambda_{rd}}$ by $E[D_{rd}^{1-sw}]$.

Gradually introducing further relaying steps amounts to recursively applying the process. The sequence of values $E[D_{sd}^{1-sw}]$, $E[D_{sd}^{2-sw}]$, ..., $E[D_{sd}^{n-sw}]$ thus created is decreasing and positive (see Sec. 3.8.2), so it converges to, let's say, $E[D_{sd}^{sw*}]$. $E[D_{sd}^{sw*}]$ is necessarily attained in a finite number of steps (since there are only a finite number of possible intermediate nodes) and is a fixed point for the recursive process. Because the set R_{sd}^* realises the fixed point, the forwarding strategy simply amounts, for node r, to relaying any message with destination d to any first encountered neighbor in R_{rd}^*.

The major benefits of the SW* routing scheme are:

i) opportunistic forwarding of messages is loop-free

ii) computing the routes is polynomial in the number of nodes in the network

Forwarding a message with SW* is loop free. Remember that a message with destination d is transfered by any node i to the first node j in set R^*_{id} it meets. The reason why there is no loop is because transfers always go to nodes that are strictly closer to the destination. All next relay nodes j have an expected delivery time to d strictly lower than that of the current relay node i (see Sec. 3.8.3). This implies that for any route a bundle takes, the sequence of nodes it visits, $i, i_1, i_2, ..., i_p$, necessarily verifies $E[D^{sw*}_{id}] < E[D^{sw*}_{i_1 d}] < ... < E[D^{sw*}_{i_p d}]$. If there were a loop, it would mean that one of the visited nodes $i_1, ..., i_p$ is i, so we would have $E[D^{sw*}_{id}] < ... < E[D^{sw*}_{i}]$, which is a contradiction.

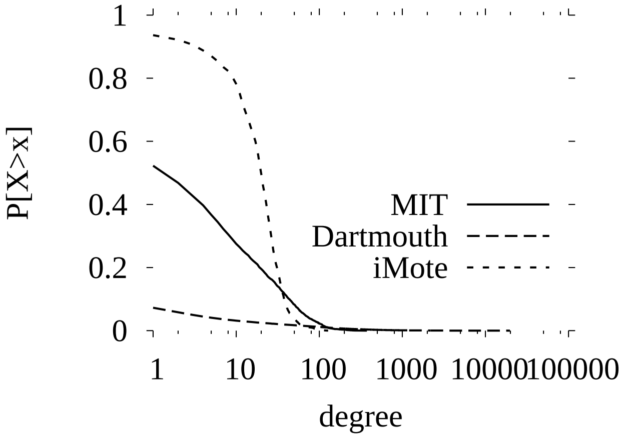


Figure 3.7: CDF of node degrees in data sets.

This algorithm has complexity of $O(L.n^2.D)$. L is the width of the *binary connectivity graph* in which two nodes share a link whenever they have been in contact. n is the total number of nodes and D is the average node degree. On the data sets we considered, L is equal to 10 for Dartmouth, 5 for MIT, and 3 for iMote. Nodes degree cumulative distributions are depicted in Fig. 3.7. We see that node spectrum of interactions are wider with respect to the network size in iMote than in MIT and Dartmouth. The average node degree is 60.5 in Dartmouth, 22.3 in MIT and 22.8 in iMote. As, in reality, D and L are very small compared to n^2, the complexity is of $O(n^2)$. More generally, if connectivity graphs have scale free properties, as one can expect in large networks, we would have $L = log(n)$ and $D \ll n$.

3.5 Comparing routing protocols

This section looks at the routing performance of a number of routing protocols, including the ones just described, in the presence of heterogeneity in inter-contact time distributions.

3.5.1 Methodology

We performed simulations using Dartmouth, iMote and MIT traces to study how the minimum delay routing algorithms behave in the case of heterogeneous connectivity in comparison to some well known approaches.

We simulate the following protocols:

- *Wait*: a node waits to meet the destination in order to transfer its message. The main advantage of this method is that it involves only one transmission per message.
- *1-SW*: unless it meets the destination, the source transmits a copy of the message it carries to the first node met. This node is used as a relay but only transmits the message to the destination if encountered.
- *1-SW**: this scheme is similar to 1-SW but relays are only chosen among the set of nodes R that minimizes Eq. 3.9, as seen in Sec. 3.4.2. We simulated this modified version of 1-SW to see if there is an advantage to taking into account the heterogeneity of contact patterns.
- MED (Minimum Expected Delay): this scheme was introduced by Jain et al. [23]. The strategy, similar to source routing, defines which path the message will follow from s to d, that is, the ordered list of intermediate relay nodes it will have to go through. The list is chosen to provide minimum expected end-to-end delay. Each relay node, upon receiving the message, will not be free to choose the next relay: it will have to follow the initial plan. Finding the optimal path thus amounts to finding a lowest-weight path between nodes s and d in a graph in which the weight on each link (i,j) is defined as $1/\lambda_{ij}$. Dijkstra's algorithm is used.
- *SW**: this is the multi-hop and single-copy routing strategy that minimizes the delay, as described in Sec. 3.4.2.
- *Epidemic*: this scheme is described by Vahdat and Becker [25]: each time two nodes meet, they exchange their messages. The major interest of this algorithm is that it provides the optimum path and thus the minimum delay. We use it here as a lower bound.

We slightly modified 1-SW, to better compare it with 1-SW*: a node i is a potential relay only if $\lambda_{id} > 0$, i.e., if it has a chance of meeting the destination. In MED, we authorized intermediary relays to directly transfer messages to the destination whenever met.

In each of the simulation series, we choose at random 100 different source destination pairs (s,d) and replay the contacts between nodes present in the data to see how, for each pair, a message, generated at the beginning of the two months period, is delivered. For each data set, because of computational issues, we used a different constant message generation rate between source destination pairs.

λ values used for route selection in 1-SW*, SW* and MED, and to determine theoretical delays for Wait, 1-SW*, SW* and MED have been computed over the whole data set in a preliminary step.

3.5.2 Simulation results

This section presents results for the different routing strategies on each of the data sets.

3.5.2.1 Dartmouth

In Dartmouth, λ values used have been computed over the data filtered due to the ping-pong effect (see Sec. 3.2.1). However, the contacts replayed in simulations were those in the original traces, as it does not impact the results and because filtering was only of interest for modelling. In these simulations, we generated messages between source-destination pairs every 20 days.

	Del. (%)	**A. delay** (days)	**M. delay** (days)	**Th. delay** (days)	**A. hops** (#)	**Overhead** (trans.)
Wait	8.6 ±1.0	12.2 ±2.7	7.2 ±4.4	11.9 ±3.1	1.0 ±0.0	25.8 ±3.1
1-SW	57.4 ±2.0	16.5 ±0.7	14.0 ±1.6	-	1.9 ±0.0	427.8 ±15.3
1-SW*	61.4 ±1.1	13.5 ±0.6	10.0 ±0.9	8.4 ±0.6	1.9 ±0.1	416.8 ±12.8
MED	34.2 ±1.2	17.9 ±1.0	15.2 ±1.8	1.0 ±0.1	6.1 ±0.2	724.8 ±20.4
SW*	82.4 ±1.4	7.8 ±0.4	4.3 ±0.3	1.4 ±0.1	5.7 ±0.1	1993.6 ±793.4
Epid.	99.0 ±0.8	1.0 ±0.2	0.9 ±0.0	-	9.8 ±0.2	123851 ±3687.8

Table 3.2: Simulation results with Dartmouth data.

Table 3.2 presents the simulation results averaged over 5 runs with 90% confidence levels that are obtained using the Student t distribution. It presents, for each of the protocols, the average delivery ratio, the average delay ("A delay") and the median delay ("M delay") computed over the delivered messages, the average theoretical delay over all the messages generated (infinite delay is assumed to be the length of the simulated period, i.e., 45 days), and the average hop count, also obtained on delivered messages. We also measured the protocol overhead, considering the total number of transmissions that occurred before message delivery (or nondelivery for those that never reached their destination).

Wait and Epidemic are the two extreme schemes that we simulated. They respectively deliver 8.6% and 99.0% of messages with a mean delay of 12.2 and 1.0 days and with a median delay of 7.2 and 0.9 days. Wait only delivers 8.6% of messages because most of the source-destination pairs, selected at random, satisfy $\lambda_{sd} = 0$ (i.e., they never met). Wait only involves 1.0 hop while Epidemic attains a high average hop-count of 9.8. Naturally, Epidemic plots the highest overhead with 123,851 transmissions in total while Wait only realizes 25.8 transmissions.

1-SW and 1-SW*, which are the two one-relay algorithms that we simulated, deliver respectively 57.4% and 61.4% of messages with an average delay of 16.5 and 13.5 days. 1-SW* outperforms 1-SW while only requiring 416.8 transmissions instead of 427.8 on average. From these results, we clearly see that one should take into account heterogeneity for routing. The difference between the modified 1-SW and 1-SW* gives further insight into the type of heterogeneity that should be considered. The modified 1-SW is a one hop strategy that uses only true relays to the destination: relay nodes in 1-SW must meet both the source and the destination. The scheme is not completely ignorant of heterogeneity, as it exploits binary connectivity information, the fact that not all nodes meet one another. 1-SW* goes beyond that and differentiates between neighboring nodes based on the quantitative expected inter-contact time. The fact that 1-SW* outperforms the modified 1-SW thus indicates that routing actually benefits from the quantitative inter-contact time heterogeneity, and not just from node connectivity.

These results show also that SW* outperforms MED while delivering 82.4% of messages instead of 34.2%. SW* has performance close to that of Epidemic in delivery ratio while only involving 1,993 transmissions. The opportunistic nature of SW* is the main reason for this superiority over MED, in which messages follow a strict sequence of relays, in a network which is not a perfect exponential DTN. A node cannot take advantage of an opportunistic contact with a node that has a lower cost path than does the predesignated next hop node. This weakness has already been mentioned by Jain et al. [23].

Table 3.2 shows a discrepancy between the theoretical and the experimental delays. This can be explained by the presence of node pairs that do not have an exponential behavior. This is particularly true for 1-SW*, SW* and MED that should show average theoretical delays of respectively 8.4, 1.4 and 1.0 days while they achieve in Dartmouth 13.5, 7.8 and 17.9 days. In this case the computation of expected delays on mean inter-contact times misses possible inter-dependencies of node contacts.

3.5.2.2 iMote

In simulations with the iMote data set, we generated messages between source destination pairs every 5 hours. Table 3.3 shows the simulation results.

We first observe that the delivery ratios are closer to each other varying from 81.9% for Wait and to 91.8% for Epidemic. The fact that Wait delivers a large number of messages is another illustration of the high level of interactions that occurred between participants, as

	Del. (%)	**A. delay** (h)	**M. delay** (h)	**T. delay** (h)	**Hops** (#)	**Overhead** (trans.)
Wait	81.9 ±2.8	10.5 ±0.6	7.2 ±0.3	5.3 ±0.5	1.0 ±0.0	1146.6 ±39.6
1-SW	83.5 ±1.2	10.6 ±0.6	7.5 ±0.6	-	1.9 ±0.0	2476.4 ±16.6
1-SW*	87.2 ±1.3	9.0 ±0.6	6.3 ±0.5	2.0 ±0.1	1.7 ±0.0	2255.0 ±34.8
MED	82.1 ±3.4	10.3 ±0.5	7.3 ±0.1	2.8 ±0.1	1.3 ±0.0	1669.6 ±31.2
SW*	88.3 ±1.4	8.6 ±0.6	6.1 ±0.7	1.7 ±0.1	2.7 ±0.1	3644.2 ±96.6
Epidemic	91.8 ±1.3	6.5 ±0.4	4.2 ±0.3	-	4.1 ±0.1	27470.6 ±950.8

Table 3.3: Simulation results with iMote data.

already seen in Sec. 3.2. We observe similar results to those with Dartmouth in ranking of protocol performance.

3.5.2.3 MIT

In the simulations we performed on MIT, messages between sources and destinations were generated every 15 days. Table 3.4 shows the simulation results.

	Del. (%)	**A. delay** (days)	**M. delay** (days)	**T. delay** (days)	**Hops** (#)	**Overhead** (trans.)
Wait	35.6 ±3.6	15.0 ±2.0	4.9 ±1.6	9.15 ±1.2	1.0 ±0.0	249.4 ±25.4
1-SW	67.7 ±2.4	11.2 ±0.5	0.8 ±0.6	-	1.8 ±0.1	1185.6 ±15.5
1-SW*	88.0 ±1.1	10.0 ±0.7	2.3 ±0.6	3.6 ±0.2	1.8 ±0.1	1080.2 ±14.5
MED	46.6 ±4.0	14.6 ±1.0	3.2 ±0.8	3.0 ±0.1	1.5 ±0.1	633.8 ±39.7
SW*	96.4 ±0.3	5.0 ±0.4	0.1 ±0.1	2.2 ±0.1	2.8 ±0.1	1994.6 ±65.5
Epidemic	99.0 ±0.2	1.4 ±0.4	0.1 ±0.1	-	2.5 ±0.1	50344.6 ±897.7

Table 3.4: Simulation results with MIT data.

Results are closer to the ones we obtained with Dartmouth. Furthermore, we observe similar ranking of protocol performance to those with Dartmouth and iMote.

Through all these simulations, we validate the natural sense that we should take into account the heterogeneity of inter-contact times distributions in the design of routing solutions for DTNs.

3.5.3 Discussion

This section discusses specific factors that could have impacted the results, and some implementation choices.

3.5.3.1 Impact of traffic generation

The results that we presented show performance that we believe to be underestimated because of the way we generated traffic. In our simulations, as we did not have any knowledge of social relationships between participants, we selected source destination pairs at random and generated traffic with a constant rate. However, in a real deployment of DTN applications, we conjecture that those two parameters would be highly driven by social relationships (most of people would only communicate with friends with who they might also have a high level of interactions) and environmental factors such as specific events or periodic schedules. As a consequence, we believe that the results that we presented might improve in a real case.

3.5.3.2 Data sets used

The data sets may represent partial or biased real life interactions as sampling methods were used for their collection. The iMote and MIT data sets have been collected using periodic Bluetooth scans which may have underestimated the overall number of contacts or the contact times between nodes. In Dartmouth, the two main factors coming into play (see Sec. 3.2) are: 1) we infer that too people are in contact whenever they are connected to the same AP which might create unrealistic interactions, 2) mobility of laptops is not really representative of human mobility. As a consequence, one has to take carefully these results into account. We conjecture, that because of those sampling methods, results are underestimated compare those we might get with perfect real connectivity data.

3.5.3.3 Complexity of inter-contact times processes

Furthermore, evaluating schemes that use synthetic information such as the average inter-contact times on real data have to deal with two factors: the presumed stationary of inter-contact processes and the short and long terms dependencies in interactions between nodes. As a consequence, average values might not be representative because processes are not stable and their burstiness is not well taking into account. An easy observation we can make to illustrate this problem is that the standard deviation for inter-contact times varies a lot for a given average. This can be observed in Fig. 3.8 where we plot the standard deviation function of the average in MIT data. For instance, for an average of 2 ± 0.2 days, standard deviations vary from 1.97 to 10.5 days.

Simulation artifacts also come into play. The routing simulation is carried out on a limited time scale. The λ values are computed over the entire data set in a prior pass, so a relay node may meet the destination for the last time before having met the source for the first time. This pre-computation being not realistic, we could have used on-line predictive or learning methods.

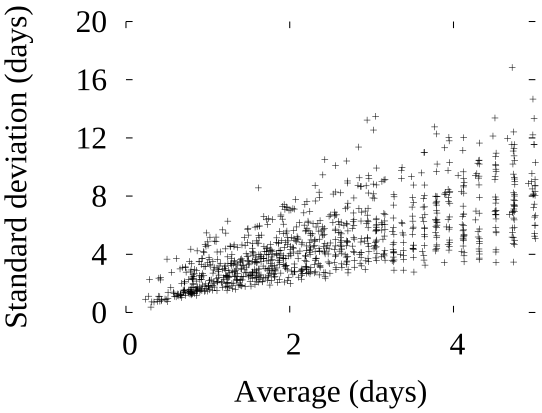


Figure 3.8: Standard deviations function of average for pairwise inter-contact times processes in MIT data.

However, as they are challenging to define, we let this study for future work and intend here to provide early validation results to motivate research in the domain. MED clearly suffers from this simulation artefact, it would have a larger number of messages otherwise.

3.5.3.4 Keeping copies at source nodes

Applications that might send data over DTN networks would probably keep a copy of messages until they get an acknowledgment telling that messages have been correctly delivered. In that case, source nodes would transfer them directly to destination nodes if encountered.

In order to study the impact of such behavior on the routing schemes that we evaluated in Sec. 3.5.2, we have performed simulations using exactly the same parameters and source destination pairs. Table. 3.5 presents results with the three connectivity data sets. We can see that keeping one copy at the source almost preserves the relative order observed previously while it slightly improves performance globally. 1-SW, 1-SW*, MED and SW* deliver in Dartmouth respectively 58.3%, 61.6%, 34.8% and 82.7% of messages instead of 57.4%, 61.4%, 34.2% and 82.4% for instance.

3.5.3.5 Overhead reduction

Handling information on contact patterns for SW* could lead to high processing and network overhead even if only synthetic information such as λ values is used. Nodes would have to perform tasks such as: monitoring the inter-contact times they have with the others, disseminating this information to the other nodes (using a centralized architecture or not), computing

	Dartmouth		**iMote**		**MIT**	
	Del. (%)	**A. delay** (days)	**Del.** (%)	**A. delay** (h)	**Del.** (%)	**A. delay** (h)
1-SW	58.3 ±1.9	16.0 ±0.8	88.9 ±1.4	8.8 ±0.5	73.8 ±2.6	10.5 ±0.6
1-SW*	61.6 ±1.1	13.5 ±0.6	88.7 ±1.3	8.7 ±0.6	88.3 ±1.3	9.7 ±0.8
MED	34.8 ±1.2	17.9 ±1.0	84.5 ±3.0	10.0 ±0.4	48.7 ±3.7	15.1 ±1.2
SW*	82.7 ±1.7	7.8 ±0.2	89.5 ±1.3	8.1 ±0.5	96.6 ±0.3	4.8 ±0.4

Table 3.5: Simulation results when a copy is kept at the source.

periodically the sets of relays that have to be used for forwarding. To reduce the amount of information shared among nodes, we evaluate a scenario in which nodes only diffuse λ values for pairs than a threshold L. Table 3.6 shows the simulation results obtained in similar conditions to those in Sec. 3.5.2 on iMote and MIT data sets.

iMote				**MIT**			
L (h)	**Del.** (%)	**A. delay** (h)	**M. delay** (h)	**L** (h)	**Del.** (%)	**A. delay** (days)	**M. delay** (days)
1	81.9 ±2.8	10.5 ±0.6	7.2 ±0.4	**1**	35.6 ±3.6	15.0 ±2.0	4.8 ±1.6
2	86.2 ±1.5	8.9 ±0.5	6.4 ±0.4	**24**	48.1 ±3.4	10.3 ±0.9	2.6 ±0.6
5	87.3 ±1.6	8.5 ±0.5	6.1 ±0.6	**36**	68.4 ±2.2	6.2 ±0.7	1.3 ±0.3
8	87.5 ±1.7	8.6 ±0.5	6.2 ±0.6	**72**	84.2 ±0.9	5.1 ±0.3	0.6 ±0.3
10	88.3 ±1.4	8.7 ±0.6	6.3 ±0.6	**168**	95.3 ±0.1	5.7 ±0.3	0.4 ±0.1
∞	88.3 ±1.4	8.6 ±0.6	6.1 ±0.7	∞	96.4 ±0.3	5.0 ±0.4	0.1 ±0.1

Table 3.6: Simulation results with partial knowledge.

We can see that, as expected, as we increase the threshold L performance are closer to those observe in Sec. 3.5.2 denoted by $L = \infty$ here. The value of L for which performance are reasonably degraded is 10 hours in iMote and 168 hours in MIT leading to reduction in shared routing information of respectively 9.8% and 35.2%. These figures depend on the overall density of interactions. Because in iMote, nodes had a high and less heterogeneous level of interactions, we are not able to reduce the overhead as we could do in MIT. We expect this reduction to be much higher in Dartmouth data but we were not able to perform simulations for computational reasons. This result is promising regarding the scalability of routing algorithms that would involve synthetic knowledge on pairwise contacts such as average inter-contact times.

3.6 Related work

Much ongoing research tries to understand and characterize the mobility patterns in DTNs and mobile ad-hoc networks (MANETs). Due to the limited number of data sets available and the fact that they are generally specific to a scenario, studies often resort to synthetic models. Models such as Random Walk (i.e., Brownian motion [56]) and Random Way-Point [57] have been very popular [30, 77]. More recent work has extended these initial models with proposals to better match patterns observed in real mobility data. Musolesi et al. [59] propose a model in which the movements of nodes are driven by social relationships. Bohacek [60] designed a mobility model of individuals in urban settings based on a recent US Bureau of Labor Statistics time-use study. Francois et al. [85] proposes a framework for formalizing the behavior contact patterns in situations in which each node knows the probability distributions for its contacts with other nodes. Carreras et al. [83] propose a graph-based model able to capture the evolution of the connectivity between nodes over time.

The approach taken in our work is rather to put the stress on inter-contact patterns as one of the key enablers for the design of routing algorithms in DTNs. This work is the first to provide a detailed analysis of the pairwise inter-contacts in a number of DTN data sets, and the first to identify the log-normal family of distributions as a promising modeling candidate.

Previous formal analyses of DTNs have focused on naive routing algorithms, such as Wait, Epidemic, or Spray and Wait routing. For controlled flooding in DTNs, Spyropoulos et al. [29] evaluates the number of copies the Spray and Wait protocol should distribute in the case of Random Walk mobility. Zhang et al. [86] use ordinary differential equations (ODEs) to investigate how resources such as buffer space and power can be traded for faster deliver using epidemic routing and its variations. Liao et al. [87] present analytic work on modeling various redundancy-based routing schemes for DTNs.

The present work moves one step further. We do not just analyse pre-existing routing schemes or evaluate the impact of their parameters. We derive a new opportunistic routing scheme that is proven to be the best at minimising single copy end-to-end delivery when pairwise inter-contacts are exponentially distributed.

3.7 Conclusion and future work

In this work, we argue for the wisdom of using pairwise inter-contact patterns to characterize DTNs. We have first provided a statistical study using widely-used DTN data sets in which we characterize heterogeneity of interactions between nodes. We show that pairwise inter-contact times processes, which have a great impact on routing, are heterogeneous and distributed in log-normal for a large number of node pairs. Second, we describe the power-law paradox and show that the distribution of aggregate inter-contact times can be power-law distributed while pairwise processes are exponentially distributed. Finally, we have validated the insight that taking heterogeneity into account for routing improves performance.

We presented a new routing strategy, SW*, which is capable of using only a subset of relays to improve routing performance, measured in terms of average delay. We show, by replaying real connectivity traces, that SW* achieves good performance, in terms of delivery ratio and delay, while keeping the overhead low. We also discussed factors and implementation issues that might have impacted the results.

Acknowledgments

We gratefully acknowledge David Kotz for enabling our use of wireless trace data from the CRAWDAD archive at Dartmouth College. We also thank Augustin Chaintreau for his valuable comments. This work was supported by LiP6 and Thales Communications through the laboratory Euronetlab, and the ANRT which provided the CIFRE grant 135/2004.

3.8 Annex

3.8.1 Result 1

For an exponential DTN of n nodes and with parameters (λ_{ij}), and for any matrix (ϵ_{ij}) which represents the estimated message travel time between node i and node j (given a certain routing policy), introducing an intermediate relay selected among a subset I of first encountered neighbor nodes of the source s, generates, for the delivery of a message to node d, an expected travel time $C(I)$, given by:

$$C(I) = \frac{(1 + \sum_{r \in I} \lambda_{sr} \epsilon_{rd})}{\sum_{r \in I} \lambda_{sr}} \tag{3.11}$$

Let I_{min} be one subset that minimises $C(I)$ for all subsets I of neighbors of s. We consider without loss of generality that $\epsilon_{1d} \leq \epsilon_{2d} ... \leq \epsilon_{nd}$.

We are going to establish the following, quite remarkable, result on the structure of the minimal set I_{min}:

THEOREM. If I_{min} is a subset of neighbor nodes of s that minimize Eq. 3.11, then either $I_{min} = \oslash$ or there is a $p, 1 \leq p \leq n$, for which $I_{min} = [1, 2, ..., p]$.

This result is derived from the special shape of criterion of Eq. 3.11. More precisely, we will need the following lemma:

LEMMA ϕ. Let's introduce the bivariate function $\phi(x,y) = \frac{b+xy}{a+x}$ (compare it to Eq. 3.11 to see how it comes into play), we have:

$$\forall b > 0,\, a > 0 \text{ and } \forall x, y \geq 0,\, \phi(x,y) \leq \frac{b}{a} \iff y \leq \frac{b}{a} \tag{3.12}$$

PROOF. This is straightforward to check.

Let's now proceed to the proof the main result.

PROOF. Consider neighbor nodes of s and whether they ever meet destination d or not:

i) if none of the neighbors of s ever see destination d, this means that using any of them as a relay introduces infinite delivery time, criterion (Eq. 3.11) becomes infinite. In other words none of the neighbors of s are valid relaying candidates, so $I_{min} = \emptyset$

ii) if at least one of the neighbors of s sees d, there exists a node with index m such that $\epsilon_{md} < \infty$ so $I_{min} \neq \emptyset$.

Let p be the index of the largest ϵ_{id} for nodes i in set I_{min}. We are going to show that all nodes i which satisfy $\epsilon_{id} \leq \epsilon_{pd}$ also belong to I_{min}.

Let's note ratio $C(I_{min})$ by $\frac{b}{a} = \frac{(d+\lambda_{sp}\epsilon_{pd})}{c+\lambda_{sp}}$. Since I_{min} minimizes criterion in Eq. 3.11, $\frac{b}{a} \leq \frac{d}{c}$: the second term represents the value of the criterion for I_{min} minus p, which is by definition of I_{min} suboptimal.

Rewriting the inequality $\frac{(d+\lambda_{sp}\epsilon_{pd})}{c+\lambda_{sp}} \leq \frac{d}{c}$, and from the property of ϕ in Eq. 3.12, we then have $\epsilon_{pd} \leq \frac{d}{c}$. Now:

$$\begin{aligned} \epsilon_{pd} \leq \frac{b}{a} &\iff \epsilon_{pd} \leq \frac{(d+\lambda_{sp}\epsilon_{pd})}{c+\lambda_{sp}} \\ \iff (c+\lambda_{sp})\epsilon_{pd} \leq d + \lambda_{sp}\epsilon_{pd} &\iff \epsilon_{pd} \leq \frac{d}{c} \end{aligned} \tag{3.13}$$

Suppose there exists a node m such that $\epsilon_{md} \leq \epsilon_{pd}$ and $m \notin I_{min}$. Let's add it to set I_{min}, and consider the value of the criterion for this new set of neighbors of s, $I' = I_{min} \cup m$, $C(I') = \frac{(b+\lambda_{sp}\epsilon_{md})}{a+\lambda_{sm}}$ which is lower than or equal to $\frac{b}{a}$ (from the property of ϕ in Eq. 3.12 and the fact that $\epsilon_{md} \leq \epsilon_{pd} \leq \frac{b}{a}$); I' would then perform better than I_{min} in minimising the criterion, which is in contradiction with the definition of I_{min}.

In other words, all (reordered) nodes 1 through p belong to set I_{min}, which provides the announced result. This further leads to the linear time algorithm for minimising the criterion: once sorted in the appropriate order, it suffices to add each node one after the other and stop when the criterion does not diminish anymore.

3.8.2 Result 2

We are going to show that for a given source destination pair s,d, the sequence of expected delivery times $E[D_{sd}^{n-sw^*}]$ for the strategy with n relays decreases as the number of relaying steps n increases.

Let's first introduce some notations. For a given destination d, let's consider, for all source nodes s, the sequence of values $E[D_{sd}^{n-sw^*}]$, defined recursively by:

$$\forall s \neq d,\, E[D_{sd}^{0-sw^*}] = \frac{1}{\lambda_{sd}} \text{ and } E[D_{dd}^{0-sw^*}] = 0 \text{ and} \tag{3.14}$$
$$\forall s, \forall n > 0, E[D_{sd}^{n-sw^*}] = Min_{R \subset P(n)}\left(\frac{(1 + \sum_{r \in R} \lambda_{sr} E[D_{rd}^{(n-1)-sw^*}])}{\sum_{r \in R} \lambda_{sr}}\right)$$

THEOREM. The sequence $E[D_{sd}^{n-sw^*}]$ defined in Eq. 3.14 is decreasing, i.e.:

$$\forall n \geq 0, \forall s,\, E[D_{sd}^{(n+1)-sw^*}] \leq E[D_{sd}^{[n-sw^*}] \tag{3.15}$$

PROOF. We proceed by induction on the number of relays n.
For $n = 0$, we have: $\forall s, E[D_{sd}^{0-sw^*}] = \frac{1}{\lambda_{sd}} \in \Re \cup \infty$
Two cases may occur, depending on whether s meets d or not:

- If $\lambda_{sd} = 0$, $E[D_{sd}^{1-sw^*}] \leq E[D_{sd}^{0-sw^*}] = \infty$.
- If $\lambda_{sd} \neq 0$, let's consider the one relay strategy for which R reduces to singleton d. Its delivery delay is given by $\frac{1}{\lambda_{sd}}$. By definition $E[D_{sd}^{1-sw^*}]$ gives a lower delay, so we have $E[D_{sd}^{1-sw^*}] \leq \frac{1}{\lambda_{sd}} = E[D_{sd}^{0-sw^*}]$.

So this proves the result in the case of $n = 0$.
Let's suppose that the result holds at rank $n-1$, $\forall s, E[D_{sd}^{n-sw^*}] \leq E[D_{sd}^{[n-1)-sw^*}]$.
Let's consider $E[D_{sd}^{(n+1)-sw^*}]$ for a given s,
By definition we have: $E[D_{sd}^{n-sw^*}] = \frac{(1+\sum_{r \in I_{min}^n} \lambda_{sr} E[D_{rd}^{(n-1)-sw^*}])}{\sum_{r \in I_{min}^n} \lambda_{sr}}$, for a given set I_{min}^n of neighbors of node s.

If one uses this set of nodes when introducing another relay node (i.e., at rank $n+1$), the expected delay is higher than $E[D_{sd}^{(n+1)-sw^*}]$ (by definition), so we have:

$$E[D_{sd}^{(n+1)-sw^*}] \leq \frac{(1 + \sum_{r \in I_{min}^n} \lambda_{sr} E[D_{rd}^{n-sw^*}])}{\sum_{r \in I_{min}^n} \lambda_{sr}} \tag{3.16}$$

But we have by hypothesis, $\forall i, E[D_{id}^{n-sw^*}] \leq E[D_{id}^{[n-1)-sw^*}]$, so this leads to:

$$E[D_{sd}^{(n+1)-sw^*}] \leq \frac{(1 + \sum_{r \in I_{min}^n} \lambda_{sr} E[D_{rd}^{(n-1)-sw^*}])}{\sum_{r \in I_{min}^n} \lambda_{sr}} = E[D_{sd}^{n-sw^*}] \tag{3.17}$$

and this is true for all nodes s, i.e., $\forall s, E[D_{sd}^{(n+1)-sw^*}] \leq E[D_{sd}^{[n-sw^*}]$. This is the induction hypothesis at rank $n+1$. QED.

3.8.3 Result 3

We are going to show that a message with destination d is relayed by SW* to a node with lower expected delivery time to d, i.e., we have the following.

THEOREM. For any node s following the routing strategy SW*, we have:

$$\forall r \in R_{sd}^* E[D_{rd}^{sw*}] < E[D_{sd}^{sw*}] \tag{3.18}$$

PROOF. $E[D_{sd}^{sw*}]$ is the fixed point of the $E[D_{sd}^{n-sw}]$ sequence, so it satisfies:

$$E[D_{sd}^{sw*}] = \frac{(1 + \sum_{r \in R_{sd}^*} \lambda_{sr} E[D_{rd}^{sw*}])}{\sum_{r \in R_{sd}^*} \lambda_{sr}} \tag{3.19}$$

Singling out a giving relay node r, and applying Lemma ϕ of Eq. 3.12, we have:

$$E[D_{rd}^{sw*}] \leq E[D_{sd}^{sw*}] \tag{3.20}$$

We now have to check that the inequality is strict. Singling out node r in Eq.3.19, we have:

$$E[D_{sd}^{sw*}] = \frac{(d + \lambda_{sr} E[D_{rd}^{sw*}])}{c + \lambda_{sr}} \tag{3.21}$$

It is straightforward to check that $E[D_{sd}^{sw*}] = E[D_{rd}^{sw*}]$ if and only if $E[D_{sd}^{sw*}] = \frac{d}{c}$. But $\frac{d}{c}$ corresponds to the criterion with set of neighbor nodes of s R_{sd}^* minus r, which is in contradiction with the definition of R_{sd}^*. So the inequality is strict.

Chapter 4

DTN routing in a mobility pattern space

Chapter 3 argued that one has to consider heterogeneous inter-contact time distributions for routing, we introduce in this chapter a generic formalism, MobySpace, which is able to use knowledge about node mobility or interactions with the others.

4.1 Introduction

This chapter addresses the problem of routing in delay tolerant networks (DTNs) [12]. It evaluates a scheme that we proposed in [9] that turns the problem of DTN routing into a problem of routing in a virtual space defined by the mobility patterns of nodes. We called this virtual space MobySpace. The earlier work tested the scheme with an entirely artificial scenario. By driving simulations with real mobility traces, in this chapter we validate MobySpace-based single-copy and multi-copy routing schemes in the context of ambient networks. This chapter also studies a number of important factors, such as the degree of homogeneity in the mobility of nodes, that

---

This is joint work with Vania Conan (Thales Communications) and Timur Friedman (Université Pierre et Marie Curie, LiP6–CNRS).

impact routing performance. Finally, the chapter examines the ability of nodes to learn their own mobility, which is important for the feasibility of such a scheme.

In one common DTN scenario, like the one we consider in this chapter, nodes are mobile and have wireless networking capabilities. They are able to communicate with each other only when they are within transmission range. The network suffers from frequent connectivity disruptions, making the topology intermittently and partially connected. This means that there is a very low probability that an end-to-end path exists between a given pair of nodes at a given time. End-to-end paths can exist temporarily, or may sometimes never exist, with only partial paths emerging. Due to these disruptions, regular ad-hoc networking approaches to routing and transport do not work, and new solutions must be proposed.

The Delay Tolerant Network Research Group (DTNRG) [13] has proposed an architecture [14] to support messaging that may be used by delay tolerant applications in such a context. The architecture consists mainly of the addition of an overlay, called the bundle layer, above a network's transport layer. Messages transferred in DTNs are called bundles. They are transferred in an atomic fashion between nodes using a transport protocol that ensures node-to-node reliability. These messages can be of any size. Nodes are assumed to have buffers in which they can store the bundles.

Routing is one of the very challenging open issues in DTNs, as mentioned by Jain et al. [23]. Indeed, since the network suffers from connectivity problems, MANET [2] routing algorithms such as OLSR, based on the spreading of control information, or AODV, which is on-demand, fail to achieve routing. Different approaches have to be found.

The problem of routing in DTNs is not trivial. Epidemic routing [25], studied by Vahdat and Becker, is a possible solution when nothing is known about the behavior of nodes. Since it leads to buffer overloads and inefficient use of transmission media, one would prefer to limit bundle duplication and instead use routing heuristics that can take advantage of the context. To move in such a direction, the DTN architecture defines several types of contacts: *scheduled*, *opportunistic*, and *predicted*. *Scheduled* contacts can exist, for instance, between a base station somewhere on earth and a low earth orbiting relay satellite. *Opportunistic* contacts are created simply by the presence of two entities at the same place, in a meeting that was neither scheduled nor predicted. Finally, *predicted* contacts are also not scheduled, but predictions of their existence can be made by analyzing previous observations.

The study presented in this chapter relies also on contacts that can be characterized as predicted, but the underlying concept is a more generic abstraction compared to previous work, being able to capture the interesting properties of major mobility patterns for routing.

The main contribution of this chapter is the validation of a routing scheme for DTNs that uses the formalism of a high-dimensional Euclidean space based on nodes' mobility patterns.

We show the feasibility of this concept through an example in which each dimension represents the frequency with which a node is be found in a particular location. We first present the evaluation of a single-copy routing scheme which uses this concept. Then, we extend this by investigating routing strategies, also based on MobySpace, which perform controlled flooding to achieve better performance in terms of delivery and delay while not impacting the network too much. We conduct simulations by replaying mobility traces to analyse the feasibility and comparative performance of such schemes.

The rest of this chapter is structured as follows. Sec. 4.2 describes the general concept of the mobility pattern based routing scheme, called the MobySpace. Sec. 4.3 presents the specific MobySpace we have considered for the evaluation. Sec. 4.4 presents simulation results for the single-copy routing scheme. Sec. 4.5 introduces multi-copy protocols based on MobySpace and proposes their evaluation. Sec. 4.6 presents a feasibility study and Sec. 4.7 provides an overview of related work concerning routing in DTNs. Sec. 4.8 concludes the chapter.

4.2 MobySpace: a Mobility Pattern Space

Two people having similar mobility patterns are more likely to meet each other, thus to be able to communicate. Based on this simple principle, our proposition [9] is to use the formalism of a Euclidean virtual space, that we call a *MobySpace*, as a tool to help nodes make routing decisions. These decisions rely on the notion that a node is a good candidate for taking custody of a bundle if it has a mobility pattern similar to that of the bundle's destination. Routing is done by forwarding bundles toward nodes that have mobility patterns that are more and more similar to the mobility pattern of the destination. Since in the MobySpace, the mobility pattern of a node provides its coordinates, called its *MobyPoint*, routing is done by forwarding bundles toward nodes that have their MobyPoint closer and closer to the MobyPoint of the destination. Note that the MobySpace is purely a virtual expression of the mobility patterns, and as such does not express the geographic coordinates of the nodes (GPS or otherwise). It cannot be used for geographic routing.

In this section, we describe manners in which mobility patterns can be characterized and the ways these patterns can be managed by the nodes, and we discuss possible limits and issues surrounding the overall concept.

4.2.1 Mobility pattern characterization

Since the mobility pattern of a node provides its coordinates in the MobySpace, the way in which these patterns are characterized determines the way the virtual space is constructed.

The way in which mobility patterns are characterized determines the number and the type of the dimensions of the specific MobySpace. It bears repeating that the MobySpace is not a physical space: each MobyPoint summarizes some characteristics of a node's mobility pattern. Many methods could be employed to describe a mobility pattern, but some requirements must be satisfied. We want mobility patterns to be simple to measure in order to keep them computationally inexpensive and to reduce the overhead associated with exchanging them between nodes. Furthermore, they must be relevant to routing, by helping nodes to take efficient routing decisions.

A mobility pattern could be based, for instance, upon historic information regarding contacts that the node has already had. A recent study [4] by Hui et al. has shown the interest of such mobility patterns. It highlights that contacts between people at the Infocom 2005 conference follow power-laws in terms of their duration. If we want to route a bundle from one node to another, we have an interest in taking the unevenness of the distribution into consideration. Intuitively, it could be very efficient to transmit a bundle to a relay that frequently encounters the destination. A MobySpace based on this kind of pattern would be as follows. Each possible contact is an axis, and the distance along that axis indicates an estimate of the probability of contact. Two nodes that have a similar set of contacts that they see with similar frequencies are close in this space, whereas nodes that have very different sets of contacts, or that see the same contacts but with very different frequencies, are far from each other. It seems reasonable that one would wish to pass a bundle to a node that is as close as possible to the destination in this space, because this should improve the probability that it will eventually reach the destination.

We might wish to consider an alternative space in which there is a more limited number of axes. If nodes' visits to particular locations can be tracked, then the mobility pattern of a node can be described by its visits to these locations. In this scenario, each axis represents a location, and the distance along the axis represents an estimate of the probability of finding a node at that location. We can imagine that nodes that have similar probabilities of visiting a similar set of locations are more likely to encounter each other than nodes that are very different in these respects.

Prior work [23] has demonstrated the interest of capturing temporal information as well. It is well known that network usage patterns follow diurnal and weekly cycles. We could easily imagine two nodes that visit the same locations with the same frequencies, but on different days of the week. This kind of desynchronisation could arise for instance in a campus at the scale of the hour if we consider two users each having a course in the same lecture hall the same day but not at the same time. Even so, it still might make sense to route to one node in order to reach the other, especially if there is a relay node at the commonly visited location. We can imagine ways in which the dimensional representation could capture temporal information as well. For instance, visit patterns could be translated into the frequential domain (by which we mean cyclic frequencies). A node's visits to a location could be represented by a point on a cyclic frequency axis, capturing the dominant cyclic frequency of visits, and a point on a phase axis, as well as a point on the axis already described, that represents the overall frequency (in terms of number of visits) of visiting the location.

The evaluation and the comparison of the different kinds of mobility patterns are kept for further studies. In Secs. 4.3 and 4.4, we test a MobySpace based on the frequency with which nodes find themselves in certain locations

4.2.2 Mobility pattern acquisition

A node in the network has to determine its coordinates in the MobySpace, the ones of the nodes it meets, and the ones of the destinations of the bundles it carries, in order to take appropriate routing decisions. Two problems arise: how does a node learn its own mobility pattern, and how does a node learn those of the other nodes?

There are several ways a node can learn its own mobility pattern. First, a node can learn its mobility pattern by observing its environment, e.g., by studying its contacts or its frequency of visits to different locations. If the node requires information about its current position, we can assume that particular tags are attached to each location. Alternatively, we can imagine that nodes are able to interrogate an exiting infrastructure to obtain these patterns. This infrastructure would act as a passive monitoring tool for pattern calculation. The system can be accessible anywhere in a wireless or in a wired fashion or it can be located at certain places.

Similarly, there are several ways that a node can learn the mobility patterns of other nodes. These mobility patterns could be spread in an epidemic fashion. Nodes could also spread just the most significant coordinates of their mobility patterns to reduce buffer occupancy and network resource consumption (an idea that we explore in Sec. 4.6.2). We can also imagine that nodes drop off their mobility patterns in repositories placed at strategic locations, and at the same time they update their knowledge with the content available at the repositories. We leave the study of possible solutions to future work.

4.2.3 Mobility pattern usage

As mentioned in the introduction, the mobility pattern of a node determines its coordinates in the MobySpace, i.e., the position of its associated MobyPoint. The basic idea is that bundles are forwarded to nodes having mobility patterns more and more similar to that of the destination. Formally, let U be the set of all nodes and L be the set of all locations. The MobyPoint for a node $k \in U$ is a point in an n-dimensional space, where $n = |L|$. We write $m_k = (c_{1_k}, ..., c_{n_k})$ for the MobyPoint of node k. The distance between two MobyPoints is written $d(m_i,m_j)$.

At a point in time, t, the node k will have a set of directly connected neighbors, which we write as $W_k(t) \subseteq U$. $W_k^+(t) = W_k(t) \cup \{k\}$ is the augmented neighborhood that contains k. MobySpace routing consists of either choosing one of these neighbors to receive the bundle or deciding to keep the bundle. The routing function, which we call f, chooses the neighbor that is closest to the destination, b. The decision for node k when sending a bundle to b is taken by applying the function f:

$$f(W_k^+(t),b) = \begin{cases} b \text{ if } b \subset W_k(t), \text{ else} \\ i \in W_k^+(t) : d(m_i,m_b) = \min_{j \in W_k^+(t)} d(m_j,m_b) \end{cases} \tag{4.1}$$

The choice of the distance function d used in the routing decision process is important. One straightforward choice is Euclidean distance. Examples of other distance functions can be found in [9]. We leave their comparison to future work.

4.2.4 Possible limits and issues

DTN routing in a contact space or a mobility space is based on the assumption that there will be regularities in the contacts that nodes have, or in their choices of locations to visit. There is always the possibility that we may encounter mobility patterns similar to the ones observed with random mobility models. The efficiency of the virtual space as a tool may be limited if nodes change their habits too rapidly.

Some problems could occur even if nodes have well defined mobility patterns, but their existence and nature may depend on the particularities of the space. For instance, in the Euclidean space, a bundle may reach a local maximum if a node has a mobility pattern that is the most similar in the local neighborhood to the destination node's mobility pattern, but is not sufficient for one reason or another to achieve the delivery. In the second type of space, where each dimension represents a location, it can happen if nodes visit similar places, but for timing reasons, such as being on opposite diurnal cycles, they never meet. This kind of user behavior has been observed by Henderson et al. [48] and Hui et al. [4].

The Euclidean spaces that we have discussed here are finite in terms of number of dimensions, but in practice the number of dimensions might be unbounded. This is the case, for instance, in the space we use as a case study in Sec. 4.3. Additional mechanisms must be found to allow this.

Finally, the routing scheme presented here is based on each node forwarding just a single copy of a bundle, which may be a problem in case of node failure or nodes leaving the system for extended periods of time. One may wish to introduce some redundancy into MobySpace routing. For instance, a node can be allowed to transmit a bundle up to T times if, after the first transmission, it meets other nodes having mobility patterns even more similar to that of the destination within a period P.

4.3 Frequency of visit based MobySpace

To evaluate the routing scheme based on MobySpace, we use a simple kind of space that we describe in the first part of this section. The second part introduces the mobility data that we replay for the evaluation.

4.3.1 Description

The frequency of visit based MobySpace we evaluate works as follow. Over a defined time interval, each node spends some portion (possibly zero) of that time at each of the n locations. This set of quantities is a node's mobility pattern, and is described by a MobyPoint in an n dimensional MobySpace. If we consider the frequencies to be reliable estimates of future probabilities, the coordinate of a node along the axis k is its probability of visit for the location k. All MobyPoints in a given MobySpace lie in a hyperplane, since we have:

$$\text{for any point } m_i = (c_{1_i}, ..., c_{n_i}), \sum_{k=1}^{n} c_{k_i} = 1 \tag{4.2}$$

Recent studies of the mobility of students in a campus [52, 48] or of corporate users [88], equipped with PDAs or laptops able to be connected to wireless access networks, show that they follow common mobility patterns. They show that significant aspects of the behavior can be characterized by power law distributions. Specifically, the session durations and the frequencies of the places visited by users follow power laws. This means that users typically visit a few access points frequently while visiting the others rarely, and that users may stay at few locations for long periods while visiting the others for very short periods. Henderson et al. observed [48] that 50% of users studied spent 62% of their time attached to a single access point, and this proportion decreased exponentially.

Regarding the distance function, we choose a straightforward one, the Euclidean distance:

$$d(m_i,m_j) = \sqrt{\sum_{k=1}^{n} \left(c_{k_i} - c_{k_j}\right)^2} \tag{4.3}$$

4.3.2 Real mobility data used

There has been considerable growth in the number of small devices people carry every day, such as cell phones, PDAs, music players, and game consoles. The variety of their different networking capabilities allows us to envisage new applications, such as distributed databases, content delivery systems, or self organizing peer to peer networks. We can imagine that such spontaneous and autonomous networks spring up around the movement of people in campus or corporate environments. Contextual applications, services, or basic applications like text messaging could take advantage of such an infrastructure. These scenarios are studied within the framework of delay tolerant networks.

For the purpose of this study, we sought real mobility traces that resembled what one might find in an ambient network environment. Since there are very few traces of this kind, we chose data that tracks mobile users in a campus setting. We used the mobility data collected on the Wi-Fi campus network of Dartmouth College [48]. Jones et al. [40] have recently used the traces in a similar way. The Dartmouth data is the most extensive data collection available that covers a large wireless access network. The network is composed of about 550 access points (APs), the number of different wireless cards (MAC addresses) seen by the network is about $13{,}000$ and the data have been collected between the years 2001 and 2004. The network covers the college's academic buildings, the library, the sport infrastructures, the administrative buildings and the student residences. Users are equipped with devices such as PDAs, laptops, and phones that support voice over IP (VoIP). The majority of the end users are students, who make intensive use of the network, especially since many of them are required to own a laptop. Fig. 4.1 illustrates the usage levels by showing the evolution of the number of active nodes in the network per day.

The data we analysed track users' sessions in the wireless network. These data have been pre-processed by Song et al. in their prior work [80] on mobility prediction. The traces show the time at which a node associates or dissociates from an access point. Data were collected by a central server with the Syslog [89] protocol. It could happen that a node does not send a dissociation message, or that a Syslog UDP message is lost, in which case a session is considered finished after 30 minutes of inactivity.

For our study, each access point represents a location. We assume that two nodes (represented as networking cards in the data) are assumed to be able to communicate with a low range device (using Bluetooth for instance), if they are attached at the same time to the same

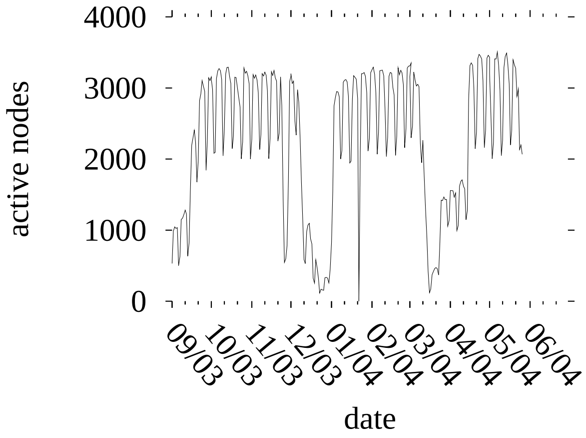


Figure 4.1: Number of actives users per day (from 1 September 2003 to 1 June 2004).

AP. This assumption is somewhat artificial as nodes that are attached to two different APs that are close to each other might be able to communicate directly. Similarly, two nodes connected to the same AP might be out of range of each other. Nonetheless, this is the best approximation we can make with the data at hand.

Though at present there are few extensive and publicly available data sets that offer mobility traces related to DTN scenarios, the situation should improve shortly. We expect for instance to evaluate MobySpace with the help of data sets like the one acquired with iMotes [4] within Intel's Haggle project or the one of the Reality Mining project [90] captured with mobile phones. These data sets provide information about fine-grained interactions between people instead of their co-presence in a coarse-grained area. Traces such as the one from the UMassDieselNet [67] project with mobile nodes on buses may also be of interest.

4.4 Simulation results

This section presents the manner in which we evaluated the routing scheme that uses a frequency of visit based MobySpace, and the results we obtained. Since we performed the simulations using a subset of 45 days of mobility data, we first describe the properties of the traces collected during that period.

4.4.1 Mobility traces

We replayed the mobility traces inferred from Dartmouth data between January 26th 2004 and March 11th 2004. Fig. 4.2 shows distributions that characterize users' behavior within this period. We choose that period because, as shown in Fig. 4.2(b), users make an intensive and regular use of the network. As shown by Fig. 4.1, this period is between Christmas and the spring break. In this period, we have observed a total of $5{,}545$ active users who have visited 536 locations.

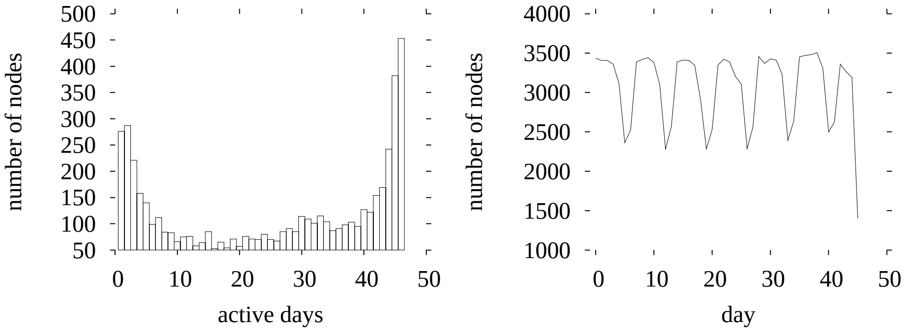
Users are mobile. They visit on average 16.66 locations in the period (see Fig. 4.2(c)) and 1.75 locations per day (see Fig. 4.2(d)). The distributions of the number of locations visited by the nodes during the period and per day follow heavy tailed distributions. This means that the majority of users have a low level of mobility while some users are very mobile. Users with a low mobility level regarding the number of locations they visit may either be users that are not very present in the data or users that stay in one place, as in students who keep their laptop connected in their room at the student residence.

The network usage displays a number of regularities. Fig. 4.2(b) shows the evolution of the number of active users per day. It highlights the existence of regular weekly cycles and a fairly constant number of active users: $2{,}901$ users per day on average. Regularity is a desirable property for this study because we wish to evaluate the MobySpace based routing scheme in a context were people move in their usual everyday environment having a number of constant habits.

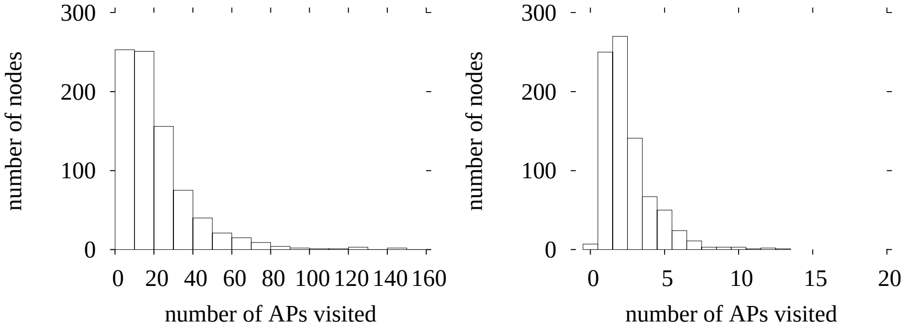
Users make intensive use of the network. The mean presence time for the period is 243 hours and is 5.18 hours per day (see Fig. 4.2(f)). Having users with a high level of presence is important but not sufficient. That presence must also be distributed over time. Thus, we analyse the distributions of the apparition and disparition days of users, and their total number of days of presence. Fig. 4.2(g) and Fig. 4.2(h) show that apparitions and disparitions generally occur close to the limits of the period. This means that the probability that a node will disappear close to the beginning of the simulation is low. Similarly, the probability that a node will appear for the first time close to the end of the period is low. Looking at the distribution of the number of days that users are present (Fig. 4.2(a)), 25.48 days on average, it appears that either users make an intensive use of their laptop or PDA, or they seldom use it, but a majority of users make an intensive use of the network since 50% of users are present more than 30 days.

4.4.2 Methodology

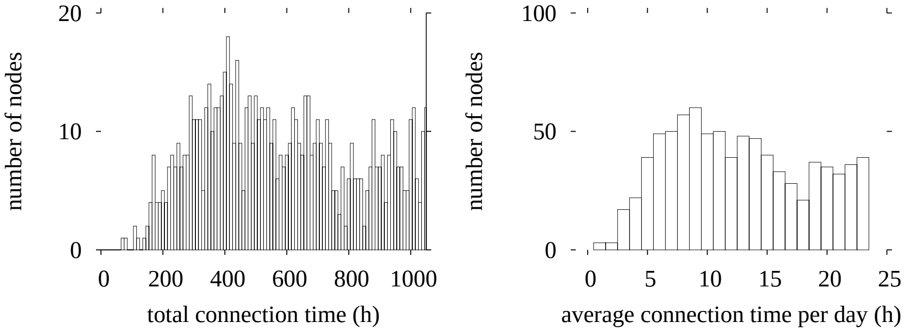
We have implemented a stand alone simulator to evaluate the routing scheme. This simulator only implements the transport and network layers and it makes simple assumptions regarding lower layers, allowing infinite bandwidth between nodes and contention free access to the medium. Nodes are also supposed to have infinite buffers and to have inherent knowledge of all other nodes' mobility patterns. Mobility patterns, which consist of sets of frequencies of visit to locations, are computed over the 45 days of data before starting the simulations. Because in ambient networks, nodes may have limited resources and capabilities, routing solutions should



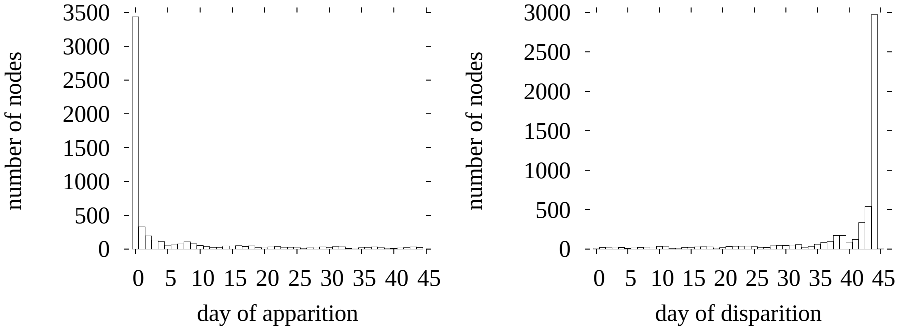
(a) Active days (b) Active nodes per day



(c) Locations visited (d) Locations visited per day



(e) Connection time (f) Connection time per day



(g) Apparition day (h) Disparition day

Figure 4.2: Statistics on the data set used for the simulations.

also be evaluated with limited buffers and more realistic models for the MAC and physical layers. One way in which we address the problem of limited resources is to examine, in Sec. 4.6.2, the possibility of limiting the amount of information that is sent regarding nodes' mobility patterns. However, our aim here is principally to validate the idea of MobySpace routing. We leave to future work a detailed study of the modifications that would be required to accommodate resource limitations. Note also that we study the question of learning mobility patterns in Sec. 4.6.

We compare the performance of MobySpace routing against the following:

- *Epidemic routing*: This is described by Vahdat and Becker [25]: Each time two nodes meet, they exchange their bundles. The major interest of this algorithm is that it provides the optimum path and thus the minimum bundle delay. We use it here as a lower bound. This algorithm can be also seen as the extension of Dijkstra's shortest path algorithm proposed by Jain et al. [23] that takes into account time-varying edge weights. In practice, epidemic routing suffers from high buffer occupancy and high bandwidth utilization.

- *Opportunistic routing*: A node waits to meet the destination in order to transfer its bundle. The main advantage of this method is that it involves only one transmission per bundle. Bundle delivery relies just on the mobility of nodes and their contact opportunities.

- *Random routing*: There are many ways to define a random routing algorithm. In order to design one that acts similarly to the MobySpace based routing scheme, we attribute for each destination node j a preference list l_j, which is a randomly ordered list of all of the nodes. When a node has a bundle destined to j, it sends that bundle to the most preferred neighbor on the preference list l_j. If the most preferred neighbor has a lower preference than the current node, the bundle is not forwarded. This mechanism avoids loops by construction.

- *Hot potato routing*: When a node is at a location and the bundle's destination in not there, the node transfers the bundle to a neighbor chosen at random. We have added a rule to avoid local loops: a node can only handle a bundle one time per location visit.

We will refer to these schemes by the following names: *Epidemic*, using Epidemic routing; *Opportunistic*, using Opportunistic routing; *Random*, using Random routing; *Potato*, using Hot potato routing, and *MobySpace*, using the routing scheme that relies on the MobySpace.

All the scenarios share common parameters that can be found in Table 4.1. We considered the whole set of 536 locations that were visited over the course of the 45 days of data.

The virtual space used for routing thus has 536 dimensions. Due to the difficulty of running simulations with the totality of the $5{,}545$ nodes, especially with Epidemic, for which computation explodes with the number of nodes and the number of bundles generated, we used a sampling method. We have defined two kinds of users: *active*, which generate traffic, and *inactive*, which only participate in the routing effort. Every active node establishes a connection towards 5 other nodes. An active node sends one bundle per connection. For active users, we chose only the ones that appear at least one time in the first week of the simulations in order to be able to study bundle propagation over an extended period. In each run, we sampled 300 users with 100 of them generating traffic. The simulator used a time step of 1s.

Parameter	**Value**
Total nodes	5545
Total locations	536
Users sampled	300
Users generating traffic	100
Simulation duration	45 days
Connections per user	5
Bundles per connection	1
Time step	1 s

Table 4.1: Simulation parameters.

We performed 5 runs for each scenario. Simulation results reported in the following tables are mean results with confidence intervals at the 90% confidence level, obtained using the Student t distribution.

4.4.3 Results

We evaluate the routing algorithms with respect to their transport layer performance. We consider a good algorithm to be one that yields a low average bundle delay, the highest bundle delivery ratio and a low average route length.

We consider two different kinds of scenarios. One with only randomly sampled users and one with only the most active.

4.4.3.1 With randomly sampled users

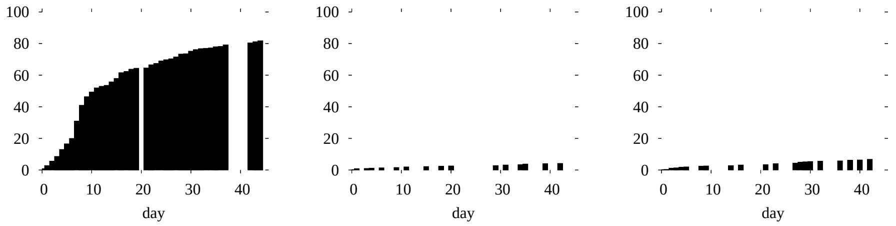
In this scenario, we picked 300 users completely at random and we replayed their traces while simulating DTN routing.

Table 4.2 shows the simulation results. It shows for each of the implemented algorithms the mean bundle delay in number of days, the mean delivery ratio, which corresponds to the number of bundles received over the number of bundles sent, and the mean route length in number of hops. The average delay and the mean route length are computed only over bundles that were delivered.

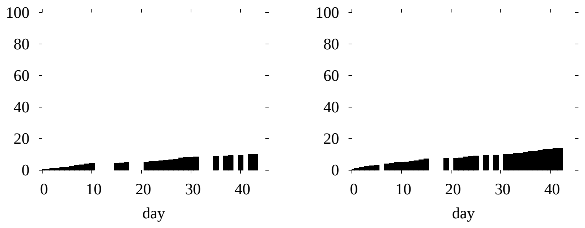
	delivery ratio (%)	**delay** (days)	**route length** (hops)
Epidemic	82.0 ±2.7	12.5 ±0.9	7.10 ±0.2
Opportunistic	4.9 ±0.6	15.9 ±2.5	1.0 ±0.0
Random	7.2 ±0.5	16.6 ±2.6	3.12 ±0.2
Potato	10.7 ±1.7	19.1 ±1.6	72.7 ±16.5
MobySpace	14.9 ±2.9	18.9 ±1.0	3.8 ±0.2

Table 4.2: Results with randomly sampled users.

The first thing we can observe is the fact that within the 45 days of simulation there is still a certain number of bundles that are not delivered with Epidemic. The mobility of the 300 nodes or their level of presence were not sufficient to ensure all the deliveries. Our sample included just 5% of the entire set of nodes. By deploying this system on more nodes, the delivery ratio would rise closer to 100%. Furthermore, we did not select nodes based on their mobility characteristics. Some of the nodes may have poor mobility.



(a) Epidemic (b) Opportunistic (c) Random



(d) Potato (e) MobySpace

Figure 4.3: Cumulative distribution of packets delivered over the 45 days (Shaded areas represent days during which packets were delivered).

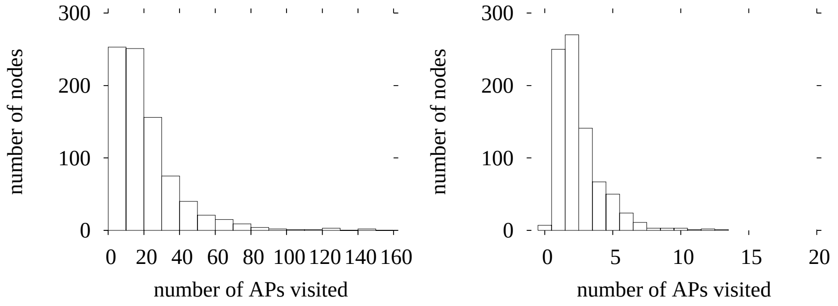
Table 4.2 shows that MobySpace delivers twice as many bundles as Random but still far less than Epidemic, which does not miss any opportunities. Random delivers somewhat more bundles than Opportunistic because the bundles are more mobile. This phenomenon is even true for Potato, which outperforms Random but delivers fewer bundles than MobySpace. At first glance, the average bundle delay of MobySpace seems poor. We believe this average is influenced by the fact that more bundles are delivered compared to the other schemes, except Epidemic. The additional bundles delivered by MobySpace might be more difficult to route than the others, leading to higher delays. The investigation of this issue is kept for future work. However, the average bundle delay is an interesting indicator of the performance an algorithm can achieve. Fig. 4.3 presents the cumulative distribution of packets delivered over time. It shows why the average bundle delay is higher for MobySpace compared to Random. It is simply because MobySpace delivers more packets in a constant fashion over time. Looking now at the average route lengths, we see that in all the cases, except Potato, they are lower than for Epidemic. MobySpace engenders routes that are about half as long as those created by Epidemic. With MobySpace, bundles are transmitted from a node to another because of their mobility patterns, not simply because of the opportunities of contact. Potato engenders routes that are extremely long because, at each contact, bundles switch from one node to another. Potato may not be suitable for a real system because of bandwidth and energy consumption issues.

4.4.3.2 With the most active users

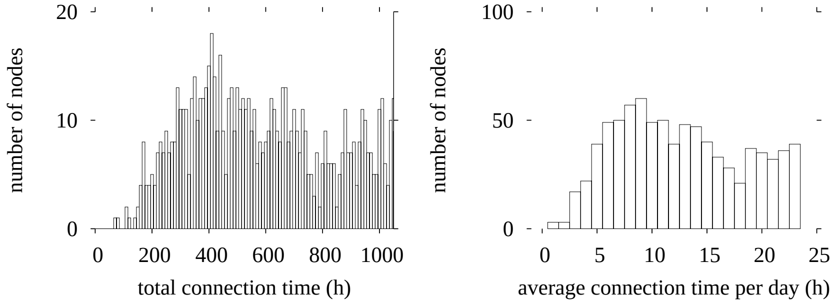
We also evaluate routing in a scenario with only the most active users, to see the effect of activity on performance. Such a scenario might also be more typical of an ambient network environment. Several metrics can characterize the level of activity. We use the regularity of the users' presence in the network, as measured by the number of active days. The number of users in our data that are active all 45 days is 835. We consider these users as a pool from which we sample for each simulation run.

As in Fig. 4.2, but here only for the most active users, Fig. 4.4 shows distributions that characterize the users' behavior. We can see that this subset of users is more active than the other. The mean presence time for the period is 609.3 hours in total and 13.13 hours per day (see Fig. 4.4(c) and Fig. 4.4(d)), as opposed to 243 hours and 5.18 hours with all the users. Users visited on average 20.65 locations in the period (see Fig. 4.4(a)) and 2.96 locations per day (see Fig. 4.4(b)), as opposed to 16.66 and 1.75 with all the users.

Table 4.3 shows the simulation results. Considering only the most active users, more bundles are delivered by the algorithms. MobySpace attains a delivery ratio of 46.6% instead of 14.9%. Note that, with respect to what we observed, the delivery ratio of MobySpace would have been higher if more nodes had participated in the scenario. However, we were limited by



(a) Locations visited (b) Locations visited per day



(c) Connection time (d) Connection time per day

Figure 4.4: Statistics on the most active users data set.

	delivery ratio (%)	**delay** (days)	**route length** (hops)
Epidemic	97.8 ±1.0	3.1 ±0.4	8.6 ±0.2
Opportunistic	10.4 ±1.4	19.6 ±1.9	1.0 ±0.0
Random	13.7 ±1.7	18.4 ±1.6	3.5 ±0.2
Potato	37.6 ±1.0	20.0 ±0.3	321.0 ±30.0
MobySpace	46.6 ±1.1	20.2 ±2.0	5.3 ±0.2

Table 4.3: Results with the most active users.

computational issues. The average bundle delay achieved is very low for Epidemic compared to the other algorithms. Route lengths are shorter than Epidemic's for Opportunistic, Random, and MobySpace, whereas the average length is higher for Potato compared to the previous scenario with randomly sampled users.

The difference between the two scenarios can be seen most clearly by looking at the delivery ratio of Epidemic. As shown by the delivery ratio of 82% obtained by Epidemic when samples are selected among all the users, the level of presence and the mobility of nodes were often not sufficient to achieve proper routing under any circumstances. Otherwise, Epidemic should have delivered 100% of the bundles. Either some of the source-destination pairs were never linked by a path over time, or certain sources and destinations were simply not sufficiently present in the data. On the other hand, when selecting only the most active users, Epidemic achieves a delivery ratio of 97.8%. The small portion of bundles not delivered comes from the fact that, in a few cases, we sampled nodes that had no interactions with the others.

These results confirm that the MobySpace evaluated in this chapter enhances routing as compared to various generic approaches for routing in an ambient network formed by users carrying personal devices in a campus setting. MobySpace achieves a high delivery ratio compared to simple algorithms like Opportunistic, Random, or Potato.

However, the proposed single-copy scheme may not achieve an acceptable delivery ratio and delay with regards to the needs of some applications. Seeing the gap that exists between the best of the single-copy schemes and Epidemic, we ask whether there might be multi-copy schemes that would allow us to approach the performance of Epidemic without the overhead associated with indiscriminate flooding. In the next section, we look at different multi-copy schemes that perform limited, or controlled, flooding, and that use MobySpace to direct the bundles that they do send.

4.5 Controlled flooding strategies

The previous section has shown that MobySpace routing in single-copy mode far outperforms the other single-copy protocols but still delivers half as many bundles as Epidemic. The question arises: can an intermediate scheme provide many of the benefits without the same overhead as Epidemic? We investigate here what would be the performance if MobySpace is used to guide a controlled flooding scheme. We describe the methodology we used for this study and we show, with simulation results, that low overhead can be achieved using MobySpace while having a delivery ratio roughly similar to that of Epidemic.

Similarly to the previous evaluations, we compare MobySpace-based controlled flooding solutions to other well-known multi-copy strategies. We first present these well-known schemes, since the ones that use MobySpace are variants:

- *Spray and Wait*: Unless it meets the destination, the source transmits a copy of the bundle it carries to the N first nodes met. These nodes are used as relays but only transmit the bundle to the destination if encountered.

- *TTL based*: The source uses a simple Epidemic scheme but with a TTL equal to T in order to only reach relays that are at most T hops away.

- *Probabilistic flooding*: The source floods its bundle like in Epidemic. However, each relay only transmits N copies to the first nodes met that do not have already it, with a probability P. Otherwise, relays act as *passive* relays like in Spray and Wait.

We will refer in the rest of the chapter to these algorithms as respectively *Spray*, *F_TTL*, *Proba*. The routing strategies based on MobySpace we evaluated are the following:

- *MobySpace Spray to Closer and Wait*: This algorithm is similar to Spray, but here, copies of a bundle are distributed only to the first N nodes met that are closer to the destination in the MobySpace. This design choice has been motivated by the fact that we want to increase the utility of each transmission.
- *MobySpace Spray to Closest and Wait*: This scheme acts similarly to *MobySpace Spray to Closer and Wait*, but copies are distributed one after another to nodes that are closer and closer to the destination in the MobySpace. The MobyPoint of the last node the bundle has been transmitted to has to be kept in memory by the source.
- *MobySpace Spray to Closer and Route*: Once distributed to relays in the same fashion as *Spray to Closer and Wait*, bundles are normally routed with the basic MobySpace single-copy scheme toward the destination.
- *MobySpace Spray to Closest and Route*: Once distributed to relays in the same fashion as *Spray to Closest and Wait*, bundles are normally routed with the basic MobySpace single-copy scheme toward the destination.
- *MobySpace Probabilistic flooding*: The difference with Proba is that when bundles are sprayed at relays, copies of bundles are distributed only to nodes being closer to the destination in the MobySpace than the current relay.
- *MobySpace Epidemic*: In this scheme, bundles are flooded like in the former Epidemic but are only transmitted from one node to another if the node is closer to the destination in the MobySpace.

We refer to these schemes as *MSpray*, *MSprayT*, *MSprayRoute*, *MSprayTRoute*, *Mproba*, *MEpidemic*.

4.5.1 Simulation results

We performed simulations exactly as in Sec. 4.4. We used the same seeds in the random number generator and the same subset of most active users that we identified in the Dartmouth data.

While in the evaluation of single-copy schemes, the overhead of protocols (except for Epidemic) was directly linked to the route lengths they induced, this is no longer the case with controlled flooding schemes. As a consequence, we use a new metric in this section, which

	N	**P**	**T**	**delivery ratio** (%)	**delay** (days)	**Overhead** (transmissions)	**route length** (hops)
Epidemic				97.8 ±1.0	3.1 ±0.4	74,674.0 ±1378.3	8.6 ±0.2
MobySpace				46.6 ±1.1	20.2 ±2.0	2,291.6 ±140.1	5.3 ±0.2
Opportunistic				10.4 ±1.4	19.6 ±1.9	52.2 ±6.9	1.0 ±0.0
Spray	5			35.5 ±2.0	18.4 ±1.4	2,554.8 ±37.8	1.8 ±0.0
	10			50.0 ±3.6	17.6 ±1.0	4,618.2 ±98.6	1.9 ±0.0
	30			67.9 ±3.4	17.8 ±0.7	8,525.2 ±107.5	1.9 ±0.0
	50			69.4 ±3.0	18.1 ±0.7	9,248.4 ±166.2	1.9 ±0.0
F_TTL			2	69.6 ±3.0	18.1 ±0.8	9,421.2 ±159.2	1.9 ±0.0
			3	93.6 ±1.8	12.2 ±1.0	25,926.8 ±874.6	2.9 ±0.0
			4	97.3 ±1.3	7.9 ±0.7	39,452.8 ±1,426.7	3.7 ±0.0
			6	97.8 ±1.0	4.6 ±0.5	55,391.0 ±1,413.7	5.2 ±0.0
Proba	2	0.1		23.6 ±2.3	19.4 ±1.3	1,334.4 ±46.7	1.8 ±0.1
	2	0.2		26.2 ±1.9	19.5 ±1.8	1,700.2 ±51.1	2.1 ±0.1
	2	0.3		31.3 ±2.0	18.4 ±1.5	2,436.0 ±98.3	2.6 ±0.1
	5	0.1		44.9 ±2.3	17.5 ±1.0	4,378.6 ±82.4	2.4 ±0.0
	5	0.2		61.6 ±2.7	15.5 ±1.0	10,911.6 ±386.0	3.6 ±0.0
	5	0.3		80.2 ±3.0	12.3 ±0.8	26,808.6 ±1,562.7	5.6 ±0.2
	5	0.6		96.2 ±1.5	5.5 ±0.5	58,492.4 ±732.2	7.8 ±0.1
	10	0.5		97.1 ±1.3	6.1 ±0.7	55,635.8 ±1209.4	6.7 ±0.1
MSpray	5			42.2 ±2.5	18.5 ±0.7	2,344.8 ±58.4	1.9 ±0.0
	10			53.6 ±2.8	18.4 ±0.7	3,785.2 ±98.0	1.9 ±0.0
	30			63.0 ±3.9	19.2 ±0.8	5,380.0 ±188.1	1.9 ±0.0
	50			63.1 ±3.9	19.2 ±0.8	5,442.8 ±194.1	1.9 ±0.0
MSprayT	5			43.6 ±1.9	19.9 ±0.7	1,743.0 ±64.9	1.9 ±0.0
	10			44.9 ±1.9	20.1 ±0.7	1,814.0 ±77.9	1.9 ±0.0
MSprayRoute	5			80.8 ±4.1	15.4 ±1.0	7,816.4 ±269.6	4.9 ±0.0
	10			85.5 ±3.3	13.7 ±1.1	11,853.6 ±424.4	4.7 ±0.0
	30			88.6 ±2.7	13.2 ±1.0	17,047.4 ±763.8	4.5 ±0.1
	50			88.7 ±2.7	13.2 ±1.0	17,921.2 ±784.0	4.5 ±0.1
MSprayTRoute	5			71.7 ±3.4	17.4 ±0.9	4,507.8 ±211.8	4.3 ±0.1
	10			72.4 ±3.5	17.4 ±0.9	4,547.6 ±218.7	4.3 ±0.1
MProba	2	0.1		31.0 ±2.6	19.1 ±1.2	1,291.0 ±36.5	2.0 ±0.1
	2	0.2		34.6 ±3.2	19.2 ±1.0	1,526.0 ±36.3	2.2 ±0.1
	2	0.3		40.8 ±2.5	19.0 ±1.2	1,908.4 ±50.0	2.5 ±0.1
	5	0.1		51.3 ±3.1	18.6 ±0.8	3,235.0 ±82.7	2.3 ±0.0
	5	0.2		61.5 ±2.9	17.6 ±1.0	4,473.0 ±106.9	2.7 ±0.1
	5	0.3		70.4 ±2.0	16.9 ±1.0	5,819.8 ±192.1	3.0 ±0.1
	5	0.6		87.6 ±2.4	13.6 ±1.0	10,154.0 ±465.7	3.9 ±0.1
	10	0.5		87.5 ±2.5	13.4 ±1.0	11,590.6 ±743.0	3.6 ±0.0
MFlooding				92.8 ±1.6	9.9 ±1.2	15,140.2 ±981.9	4.4 ±0.1

Table 4.4: Simulation results for controlled flooding schemes.

is the total number of transmissions that occurred before bundle delivery (or non delivery for those that never reached their destination).

Table 4.4 presents the simulation results. This table is divided into three parts: (1) results for the main protocols we evaluated in Sec. 4.4, (2) results for the usual multi-copy algorithms and (3) results for MobySpace-driven controlled flooding solutions.

Epidemic and Opportunistic show the two extremes. The first one delivers 97.8% of bundles with an average delay of 3.1 days, an average route length of 8.6 and $74{,}674.0$ transmissions. The second one only delivers 10.4% of bundles with an average delay of 19.6 days, an average route length of 1.0 and 52.2 transmissions. As seen previously, the delivery ratio for MobySpace is 46.6% which is right in between these two extremes but with an overhead closer to Opportunistic with $2{,}291.6$ transmissions.

Looking at the well-know multi-copy schemes we evaluated, we observe that the higher the number of copies sent into the network, the better the delivery ratio and the higher the overhead. None of these solutions outperforms MobySpace in delivery ratio; they all lead to a higher overhead for an equivalent delivery ratio. For instance, Spray with $N = 5$ achieves 50.0% delivery but with an overhead of $4{,}618.2$ which is more than twice as much as MobySpace.

In some of the cases, MobySpace-based controlled flooding solutions improve delivery ratio while leading to lower overhead. For instance, with 5 copies distributed Spray achieves 35.5% delivery with $2{,}554.8$ transmissions while MSpray obtains 42.2% delivery with only $2{,}344.8$ transmissions.

In other cases, especially when the number of copies distributed is high (e.g., Spray and MSpray with $N = 50{,}100$, or Proba and MProba with $N = 5$, $P = 0.6$), the MobySpace-based solutions show a lower delivery ratio but lead to a significantly reduced overhead. MSpray with $N = 5$ leads to half as much overhead while delivering only 6.3% fewer bundles. This is due to a lack of opportunism of MobySpace based schemes in their forwarding decisions. The average delay suffers also from this lack. The delay for MProba with $N = 5$ and $P = 0.6$ is 8.1 days higher than Proba with the same parameters.

MSprayRoute shows encouraging performance. With only 5 copies distributed, it achieves 80.8% of delivery with an overhead of only $7{,}816.4$. The two variants MSprayT and MSprayRouteT have lower overheads compared to their homologous MSPray and MSprayRoute but with, as expected, lower delivery ratios. Note also that they were only feasible with $N = 5$ and $N = 10$ because they were not able to distribute 30 copies. There were only a few opportunities for sources to transfer bundles to nodes that are closer and closer to the destination in the MobySpace.

MFlooding has one of the best results, it delivers 92.8% of bundles with an average delay of 9.9 days while only using $15{,}140.2$ transmissions (80% less than Epidemic).

Fig. 4.5 highlights the trade-off that exists between the proportion of undelivered bundles and the overhead for each class of protocols we evaluated. We see that the trade-off takes a concave form, going from the upper left part with Opportunistic to the lower right part with Epidemic. Since our goal is to minimize both the overhead and the number of bundles not

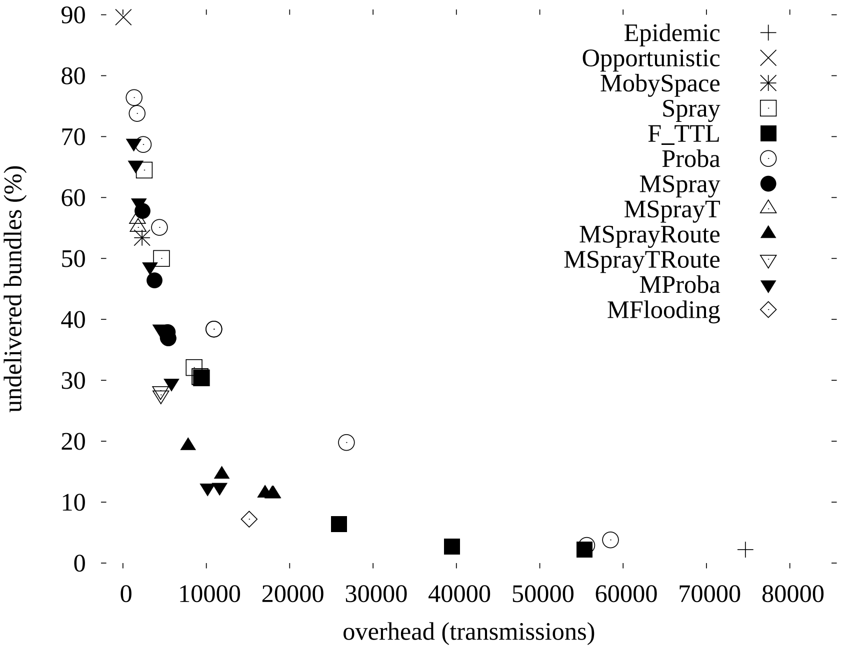


Figure 4.5: Trade-off between delivery and overhead.

delivered, this plot shows that MobySpace-based solutions such as MSprayRoute with $N = 5$ and MFlooding, which are in the bend of the curve, tend to reach our main objective.

These results have shown that allowing multiple copies to be sent, while maintaining MobySpace's primary objective, which is to get closer at each transmission to the destination in the MobySpace, is a real benefit for the performance of DTN routing schemes.

However, as we have seen, the kind of knowledge about node mobility that we used for routing does not allow us, on the data we used, to achieve the same performance as Epidemic, in delay especially. More knowledge, or knowledge that better characterizes node mobility, would certainly improve routing performance. The trade off is then to find the most relevant information and its most efficient use for routing, without impacting the network too much in terms, for instance, of computation power required by nodes or amount of information shared among nodes. Furthermore, the difference between the results obtained with MobySpace and that of Epidemic might be explained by the fact that a large number of interactions are simply not predictable because of the complexity hidden in node connectivity patterns. MobySpace-based solutions are not able to take advantage of these unpredicted interactions, while Epidemic is.

4.6 Feasibility

Previous sections have shown encouraging results for the use of MobySpace. However, the simulations rely on the assumption that nodes are aware of their mobility patterns. This section examines two different factors that impact the feasibility of this architecture: the characteristics of the mobility patterns and the possibility of learning them.

4.6.1 Mobility pattern characteristics

As noted in our prior work [9], when nodes do not have a high degree of segregation in their mobility patterns, MobySpace can not benefit from the patterns for efficient routing. We analyse here the properties of the mobility patterns we compute on users of Dartmouth College with the help of the relative entropy, S_r, applied to the set of probabilities that make up a mobility pattern. This metric describes the homogeneity of mobility patterns, which is 1 for a pattern with no preference among locations and is small for patterns that strongly prefer a few locations. It is defined for the mobility pattern of node k by:

$$S_r(k) = -\frac{\sum_{i=1}^{n} c_{i_k} \ln c_{i_k}}{\ln n}, \text{ with } n \text{ the number of dimensions} \tag{4.4}$$

The relative entropy is relevant for the analysis of mobility patterns because it captures a number of important characteristics. The relative entropy is at the same time correlated to the number of locations visited and to the time spent at each location. If a node is equally likely to be found in any location, it has the maximum relative entropy value of 1. If it is very likely to be found in one of a few locations, and unlikely to be found in any other, it has low relative entropy.

Fig. 4.6 shows the distribution of the relative entropy of users' mobility patterns for the period of 45 days. They display generally low entropy: on average 0.15. The patterns tend to demonstrate good properties for the MobySpace routing scheme because either they contain few components or they contain many components in a non homogeneous fashion.

We study the effect of pattern entropy on MobySpace routing. Table 4.5 shows that the relative entropy of mobility patterns has a great influence the performance in terms of the number of packets that are delivered. The higher the relative entropy, the higher the delivery ratio. Route lengths are stable over the increase of the relative entropy, except for Potato that generates longer routes.

These results show that a lack of diversity in the movements of users does not favor routing in such an environment. In our prior work [9] we demonstrate, with an artificial scenario, that too much diversity can also be a problem if mobility patterns can not be distinguished. In

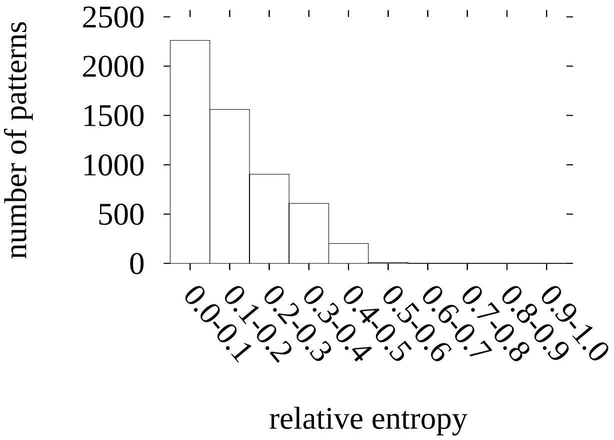


Figure 4.6: Relative entropy distribution of mobility patterns.

that case, distances in MobySpace have little significance. We were not able to reproduce this demonstration with Dartmouth data because there is no user in the data that visits almost all the locations in a regular fashion. We can conclude that a MobySpace approach is of interest when mobility patterns display a low relative entropy, but not too close to 0.

metric	S_r	**delivery ratio** (%)	**delay** (days)	**route lengths** (hops)
Epidemic	$[0.0-0.1]$	45.4 ±5.1	24.1 ±1.7	7.0 ±0.2
	$[0.1-0.2]$	79.6 ±3.2	13.1 ±1.8	8.0 ±0.4
	$[0.2-0.3]$	97.8 ±1.7	8.7 ±1.3	7.5 ±0.4
	$[0.3-0.4]$	99.0 ±0.5	6.0 ±0.9	7.1 ±0.4
Opportunistic	$[0.0-0.1]$	2.2 ±0.3	15.0 ±3.8	1.0 ±0.0
	$[0.1-0.2]$	4.4 ±0.9	19.8 ±2.4	1.0 ±0.0
	$[0.2-0.3]$	9.6 ±2.0	19.9 ±1.0	1.0 ±0.0
	$[0.3-0.4]$	24.5 ±2.5	10.9 ±0.9	1.0 ±0.0
Random	$[0.0-0.1]$	2.3 ±0.4	11.6 ±4.5	2.0 ±0.3
	$[0.1-0.2]$	5.8 ±1.2	20.0 ±2.6	3.0 ±0.2
	$[0.2-0.3]$	12.3 ±1.4	17.6 ±2.5	3.5 ±0.1
	$[0.3-0.4]$	29.5 ±3.0	12.5 ±1.1	3.9 ±0.1
Potato	$[0.0-0.1]$	3.2 ±0.8	16.9 ±1.4	43.0 ±12.0
	$[0.1-0.2]$	9.6 ±1.1	19.8 ±2.8	116.2 ±44.2
	$[0.2-0.3]$	19.8 ±5.6	20.2 ±1.5	162.7 ±44.7
	$[0.3-0.4]$	36.6 ±4.9	12.0 ±1.3	176.6 ±14.3
MobySpace	$[0.0-0.1]$	3.4 ±0.4	14.9 ±1.8	2.5 ±0.2
	$[0.1-0.2]$	8.4 ±2.4	19.5 ±2.3	3.3 ±0.2
	$[0.2-0.3]$	19.8 ±2.4	19.7 ±1.2	4.0 ±0.2
	$[0.3-0.4]$	42.3 ±4.8	13.4 ±1.3	4.7 ±0.2

Table 4.5: Results with users having different entropy.

4.6.2 Space reduction

Because transmitting nodes' entire mobility patterns is potentially expensive, we evaluate a scenario in which nodes only diffuse the main components of their mobility patterns. If we sort a node's frequencies of visit to locations in decreasing order, we mean by the main components those frequencies that are at the beginning of this list. All components not transmitted are treated as zeros. (Note that in such a case, MobyPoints no longer all lie on a hyperplane, as the sum of the frequencies can be less than one.) We ran simulations taking into account only the principal 1st, 2nd, or 3rd components of mobility patterns of nodes, and we consider the most active users.

l	**delivery ratio** (%)	**delay** (days)	**route length** (hops)
$l = 1$	39.2 ± 5.9	20.2 ± 2.6	4.9 ± 0.4
$l = 2$	46.3 ± 3.3	19.9 ± 1.2	5.2 ± 0.2
$l = 3$	47.5 ± 4.6	19.4 ± 1.8	5.2 ± 0.2
$l = 536$	50.4 ± 4.7	19.5 ± 1.3	5.1 ± 0.2

Table 4.6: Results with space reduction. l is the number of most significant components taken into account.

Table 4.6 shows that the higher the number of components taken into account, the higher the performance. Surprisingly, the delivery ratio tends very quickly to that of the scenario where all the components are used. These simulations show that only few components are needed to be exchanged between nodes in order to perform routing.

4.6.3 Mobility pattern learning

One important condition for the applicability of the MobySpace is whether users can learn their own mobility patterns. In this section we provide a first study on this issue with the Dartmouth data.

For that purpose, we split the 45 days of Dartmouth data into two periods: the learning period and the routing period. The learning period consists of the first 15 days and the routing period, the last 30 days. We study here how well the mobility patterns of nodes learnt in the learning period match the mobility patterns that characterize the routing period. The error is measured as to be the Euclidean distance d between the two mobility patterns, divided by the maximum possible distance between two mobility patterns in the hyperplane:

$$e = \frac{d}{\sqrt{n}}, \text{ with } n \text{ the number of dimensions} \tag{4.5}$$

We varied the number of days devoted to learning during the learning period, starting with the one day immediately prior to the routing period, and working back to cover all 15 days of the learning period. Fig. 4.7 shows the prediction error of mobility patterns, as a function of the number of days devoted to learning. We made this computation for all the nodes and for only the most active ones. We see that, in both cases, the longer nodes learn their own mobility, the closer their mobility patterns approximate the patterns of the routing period. As expected, the most active users learn their patterns more rapidly than the others.

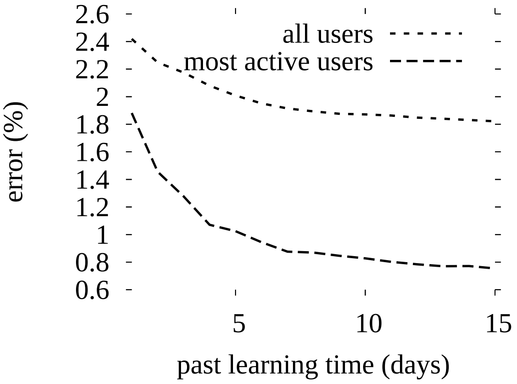


Figure 4.7: Prediction error of mobility patterns.

These initial results on the ability of nodes to learn their own mobility patterns are encouraging. They indicate that nodes might be able to benefit from their past knowledge to make routing decisions within the MobySpace. Nevertheless, further studies are needed to quantify possible long and short term dependencies in mobility traces. This must be also validated on other mobility traces.

4.7 Related work

Some work concerning routing in DTNs has been performed with scheduled contacts, such as the chapter by Jain et al. [23] about improving the connectivity of an isolated village to the Internet based on knowledge of when a low-earth orbiting relay satellite and a motor bike might be available to make the necessary connections. Also of interest, work on interplanetary networking [7, 91] uses predicted contacts such as the ones between planets within the framework of a DTN architecture.

The case of only opportunistic contacts has been analyzed by Vahdat and Becker [25] using the epidemic routing scheme that consists of flooding. The ZebraNet project [26] is exploring this idea to perform studies of animal migrations and inter-species interactions. Data are flooded in the network such as they get back to access points using animals' mobility. In order to control flooding in DTN, Spyropoulos et al. have introduced the Spray and Wait [29] protocol that distributes a number of copies to relays and then waits until the destination meets one of them. Harras et al. [30] have evaluated simple controlled message flooding schemes with heuristics based, for instance, on hop limits or timeouts. They also introduce a mechanism based on packet erasure. Once a message arrives at the destination after basic flooding, the remaining copies in the buffers of other nodes are erased. Wang et al. [31] reincode the messages with erasure codes and distribute their different parts over a large number of relays, so that the original messages can be reconstituted even if not all packets are received. Widmer et al. [32] have explored network coding techniques. All these approaches distribute multiple copies of packets, they ensure a high reliability of delivery, and a low latency, but they imply high buffer occupancy and high bandwidth consumption. Small et al. [92] propose an analytical study of existing trade-offs between resources consumption such as energy, throughput, buffers and the performance in term of latency.

Some research projects such as Data Mules [1] or SeNTD [41] use mobile network elements to transport data from fixed sensors to a number of access points in an opportunistic fashion. For instance, in SeNTD, data from sensors placed on buoys that monitor the water quality on a lake are relayed by tourist tour-boats or pleasure cruisers.

A large amount of work concerning routing in DTNs has also been performed with predicted contacts, such as the algorithm of Lindgren et al. [33], which relies on nodes having a community mobility pattern. Nodes mainly remain inside their community and sometimes visit the others. As a consequence, a node may transfer a bundle to a node that belongs to the same community as the destination. This algorithm has been designed as a possible solution to provide Internet connectivity to the Saami [93] population who live in Swedish Lapland with a yearly cycle dictated by the natural behavior of reindeer. In a similar manner, Burns et al. [36] propose a routing algorithm that uses past frequencies of contacts. Also making use of past contacts, Davis et al. [37] improved the basic epidemic scheme with the introduction of adaptive dropping policies. Recently, Musolesi et al. [38] have introduced a generic method that uses Kalman filters to combine and evaluate the multiple dimensions of the context in which nodes are in order to take routing decisions. The context is made of measurements that nodes perform periodically, which can be related to connectivity, but not necessarily. This mechanism allows network architects to define their own hierarchy among the different context attributes. LeBrun et al. [39] propose a routing algorithm for vehicular DTNs using current position and trajectories of nodes to predict their future distance to the destination. They replay GPS data collected from actual buses in the San Francisco MUNI System, through the NextBus project.

Finally, Jones et al. [40] propose a link state routing protocol for DTNs that uses the minimum expected delay as the metric.

4.8 Conclusion

The main contribution of this work has been the validation of a generic routing scheme that uses the formalism of a high-dimensional Euclidean space constructed upon mobility patterns, the MobySpace. We have shown, through the replay of real mobility traces, that it can applied to DTNs and that it can bring benefits in terms of enhanced bundle delivery and reduced communication costs. We have evaluated the use of MobySpace not only for single-copy routing, but also as a means to drive and improve existing basic controlled flooding solutions.

This chapter has also presented results of a feasibility study in order to determine the impact of the characteristics of nodes' mobility patterns on the performance and to study nodes' ability to learn their patterns. Thus, to make DTN routing work with the MobySpace, nodes need to have a minimum level of mobility with mobility patterns that can be sufficiently discriminated. We present encouraging results about the capacity of nodes to learn their own patterns. And, we also see that nodes can reduce the number of components in the mobility patterns without great impact on routing performance. This can reduce the overhead of MobySpace and the complexity of handling mobility patterns.

Acknowledgments

We gratefully acknowledge David Kotz for enabling our use of wireless trace data from the CRAWDAD archive at Dartmouth College. We thank Marc Giusti and Pierre Lafon at the STIX laboratory (École Polytechnique / CNRS) for access to the machines we used for the simulations. This work was supported by E-NEXT, an FP6 IST Network of Excellence funded by the European Commission. Also, LiP6 and Thales Communications supported this work through their joint research laboratory, Euronetlab, and the ANRT (Association Nationale de la Recherche Technique) provided the CIFRE grant 135/2004.

Chapter 5

Content distribution in an urban setting

THIS chapter presents our last contribution to DTN. It investigates the feasibility of a city-wide contentdistribution architecture composed of short range wireless accesspoints. With the analysis of the traces from an iMote experiment we conducted in the city of Cambridge, UK, we look at how a target group of intermittently and partially connected mobile nodes can improve the diffusion of information within the group by leveraging fixed and mobile nodes that are exterior to the group

5.1 Introduction

This increased penetration of wireless-capable handheld devices has led to the development of new communication techniques. Such communication techniques include *opportunistic networking*, which makes use of the capability of the devices to communicate locally among their neighbors to create communication possibilities with users and devices in other places, even

---

This is joint work with Anders Lindgren (Luleå University of Technology), James Scott (Intel Research Cambridge), Timur Friedman (Université Pierre et Marie Curie, LiP6–CNRS) and Jon Crowcroft (University of Cambridge).

when if there never exists a fully connected path between the two end-points. These networks are a type of delay tolerant network (DTN) [13] and fall also under the Pocket Switched Networking (PSN) paradigm [50]. In this context, this chapter investigates the feasibility of a city-wide content distribution architecture for electronic newspapers or local information. We look at how a target group of intermittently and partially connected mobile nodes can improve the diffusion of information within the group by leveraging various mixtures of fixed and mobile nodes that are exterior to the group. The fixed nodes are data sources, and the external mobile nodes are data relays, and we examine the trade off between the use of each in order to obtain high satisfaction within the target group, which consists of data sinks.

To evaluate the different content distribution schemes we propose, we conducted an experiment in the city of Cambridge, UK, in which 20 stationary devices equipped with a Bluetooth contact logger were deployed at popular places. We then ran simulations in which we imagined that these devices were access points distributing electronic content. In addition to this, we deployed 40 similar contact loggers on a group of students from Cambridge University. Because we used Bluetooth technology, we gathered interactions not only between the contact loggers, but also with a large number of other Bluetooth enabled devices such as mobile phones or PDAs. In our simulations, students were the target group, making the assumption that they were all interested in the content distributed by the access points, and Bluetooth devices external to the experiment could potentially be data relays. We are making the data collected in this experiment available to the research community [45]. We therefore devote a part of this chapter to a description of the salient characteristics of the dataset.

This chapter has two main contributions. First, it presents an original data set using fixed iMotes. Second, using these data, it evaluates performance of a city-wide content distributing architecture. This chapter validates the use of opportunistic networking in the particular environment we studied. It shows that despite the fact that students did not on average meet a large number of access points each day, we can achieve good performance in delivery ratio, delay and resource utilization with a content distribution scheme that allows students to collaborate. We also demonstrate that the use of Bluetooth devices external to the experiment to relay the content can make an incremental but important increase in performance in both an increased delivery ratio and a decreased delay. Finally, we investigate the robustness of the content distribution infrastructure and we show that decreasing the number of members of the target group or the number of access points increase interest of using exterior nodes as relays.

The rest of this chapter is structured as follows. 5.2 describes the experiment setup. 5.3 presents the analysis of the mobility traces that were collected. 5.4 details the content distribution schemes proposed and evaluates them. 5.5 provides an overview of related work concerning mobility data acquisition. 5.6 concludes the chapter.

5.2 Experiment setup

In the experiment we performed, we were interested in tracking contacts between different mobile users, and also contacts between mobile users and various fixed locations. Previous experiments have measured contacts between mobile users in corporate and conference settings [50] by requesting users to carry small Intel iMote[1] devices that can log contacts with other Bluetooth enabled devices. We chose to use the same technology to gather contacts. Mobile users in our experiment mainly consisted of students from Cambridge University who were asked to carry these iMotes with them at all times for the duration of the experiment. In addition to this, we deployed a number of stationary nodes in various locations that we expected many people to visit such as grocery stores, pubs, market places, and shopping centers in and around the city of Cambridge, UK. A stationary iMote was also placed at the reception of the Computer Lab, in which most of the experiment participants are students. 5.1 shows the positions of the stationary nodes. The road that rings the center of Cambridge, an area of 3 km^2, is clearly visible on the map.

Figure 5.1: Locations of fixed iMotes.

To discover other nearby users and to be able to log contacts between nodes, the iMotes use the Bluetooth inquiry mechanism that allows them to get knowledge of all other Bluetooth enabled devices within radio transmission range. As conducting the inquiries requires transmitting and receiving over the radio interface, this consumes power and a trade-off that had to be considered when setting up the experiment was how to set δ, the interval between inquiries. Indeed, having a δ too low would have lead to a shortened lifetime of the iMotes due to the high power consumption from frequent use of the radio. On the other hand, setting δ to a too

---

[1] The iMotes are small sensor platforms with an ARM7 processor and some on board storage and Bluetooth capability.

high value means running the risk of missing more potential contacts. Note that when an iMote is not inquiring, it answers to other iMotes' enquiries.

To determine the inquiry interval to use, we studied power consumption on the iMotes while idle and while performing inquiries. Using these measurements in conjunction with experience on the life-time of iMotes in previous experiments, we chose inquiry intervals that we hoped would allow the devices to have a life-time of 2 weeks. Furthermore, there is a small risk that the Bluetooth inquiry may occasionally miss a contact even though it is present. Therefore, we made the decision that if a contact is seen at a given inquiry I_i, but not at the subsequent at inquiry I_{i+1}, we will still assume that the recorded contact was never broken if we observe it again at the following inquiry I_{i+2}. This assumption was also made in previous contact logging experiments using iMotes.

iMotes carried by students had to be packaged within a small form factor to increase the probability that the users would actually always carry the device and not leave it behind. On the other hand, we had larger freedom when it came to the stationary devices. Thus, for some of the fixed iMotes, we added extra battery power to be able to reduce the inquiry interval so that we would detect more of the possible contacts. Furthermore, on a few of the fixed iMotes, we were also able to attach external antennas with greater wireless range. This increases the coverage area in which they can detect mobile devices in large public places.

- *MSR-10*: Mobile Short Range iMotes with an interval of 10 minutes between inquiries. These iMotes were given to a group of 40 students, mostly in the 3rd year at the Cambridge University Computer Lab. The devices were packaged in small boxes (dental floss boxes) to be easy to carry around in a pocket, and used a CR-2 battery (950 mAh) for power.

- *FSR-10*: Fixed Short Range iMotes with an interval of 10 minutes between inquiries. We deployed 15 of these iMotes in fixed locations such as pubs, shops or colleges' porter lodge. We used exactly the same packaging and batteries as the MSR-10.

- *FSR-6*: Fixed Short Range iMotes with an inquiry interval of 6 minutes. These iMotes were equipped with a more powerful rechargeable battery providing 2200 mAh so that we were able to reduce the inquiry interval to 6 minutes. We deployed 2 of these.

- *FLR-2*: Fixed Long Range iMotes with an interval of 2 minutes between inquiries. To increase the area in which these iMotes can discover other devices, four devices were equipped with an external antenna, which provided a communication range that was approximately twice that of the short range iMotes. Further, these iMotes were also equipped with 3 more powerful rechargeable batteries providing 2200 mAh so that we could reduced the inquiry interval to 2 minutes. Their antenna and packaging can be seen in 5.2.

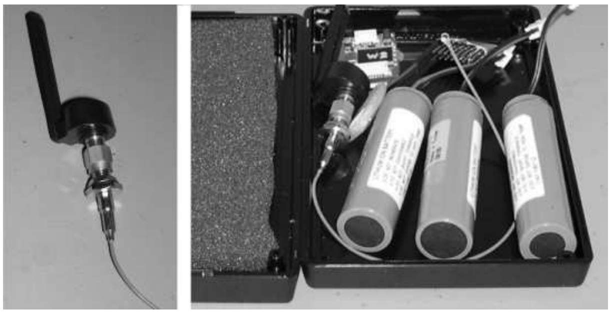


Figure 5.2: Long range iMote with rechargeable batteries.

To prevent the results from being biased by the fact that the mobile devices are co-located as they were being deployed to their carriers, we have removed the data collected during first 3 hours of the experiment from the analysis. After the mobile devices had been given to the experiment participants, we proceeded to the city centre to deploy the stationary iMotes at their respective locations. The experiment started on Friday, October 28th 2005, 9:55:32 (GMT) and stopped on Wednesday, December 21th 2005, 13:00 (GMT).

5.3 Data analysis

Due to various hardware problems and the loss of some of the deployed iMotes, we were able to gather measurement data from 36 mobile participants and 18 fixed locations, as 5.1 shows. This table presents statistics about the experiment. It shows that the average lifetimes for all types of iMotes is higher than 10 days and that these results present a low variability by type except for FLR-2. Indeed, while 2 of the FLR-2 could remain active for the full 23 days of the deployment, 2 of them that were deployed in very popular places suffered from a buffer overflow after 5 and 9 days respectively, having recorded on average 3,670 contacts.

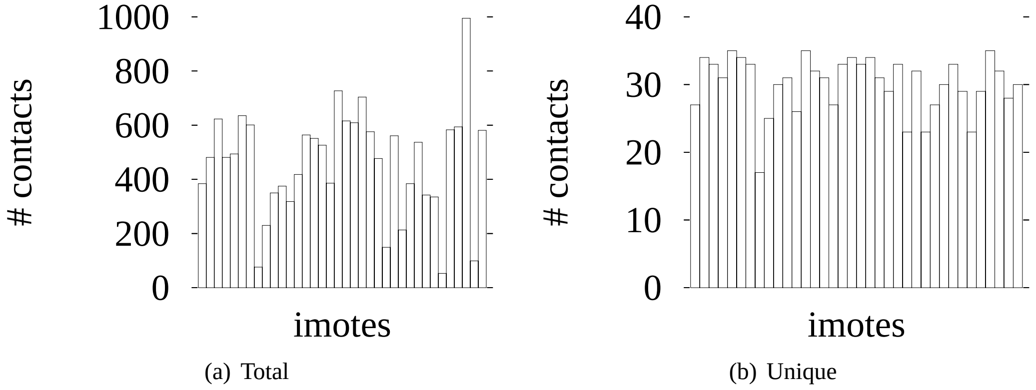
In our analysis, we consider two categories of contacts: *internal contacts*, which are contacts that occurred between two iMotes of any type (fixed or mobile), and *external contacts*, which are contacts that occurred between an iMote and another Bluetooth capable device (e.g., PDA or mobile phone). 5.1 shows the number of contacts acquired by all the types of iMotes for all the categories of contacts. The table also lists the number of *unique* contacts that has been seen in the different categories. Unique contacts are the number of different node pairs that ever have a contact over the course of the experiment duration. We can first see that, as expected, the MSR-10 iMotes had a large number of contacts with each other and that they also had a significant number of contacts with external devices, $10{,}469$ in total. The second immediate observation is that fixed iMotes had a very large number of contacts ($20{,}240$ in total) with external devices, while they did not meet the participants of the experiment very much, with only 231 contacts in total. Despite the small number of FLR-2 iMotes that were deployed, their placement at very popular locations allowed them to capture a large number of external contacts.

	MSR-10	**FSR-10**	**FSR-6**	**FLR-2**
Nb motes	36	12	2	4
Lifetime (days)	10.7 ± 0.8	11.0 ± 0.6	14.5 ± 0.5	15.7 ± 8.3
Contacts	19014	8270	1082	11119
Int. co.	8545	38	91	102
Ext. co.	10469	8232	991	11017
Contacts (u)	5681	6189	815	6789
Int. co. (u)	644	25	35	43
Ext. co. (u)	5037	6164	780	6746

Table 5.1: Global statistics. (u) means unique contacts.

5.3.1 Inter-students contacts

Here we analyse the interactions we observed between participants carrying iMotes. We first note that they had a large number of contacts together, as 5.3 shows. 5.3(a) and 5.3(b) present, respectively, for each mobile iMote, the number of total contacts and unique contacts. On average, students had 461.9 internal contacts with a standard deviation of 196.2 and 30.0 unique internal contacts over the 35 possible with a standard deviation of 4.0.



(a) Total

(b) Unique

Figure 5.3: Contacts between mobile iMotes.

5.4 shows the number of contacts per day between mobile iMotes. Most of the contacts occurred during week days, and less contacts have been recorded during Saturdays and Sundays (i.e., days 2, 3 and 9, 10). In a group of students, in which most of them belong to the same program, this observation is natural.

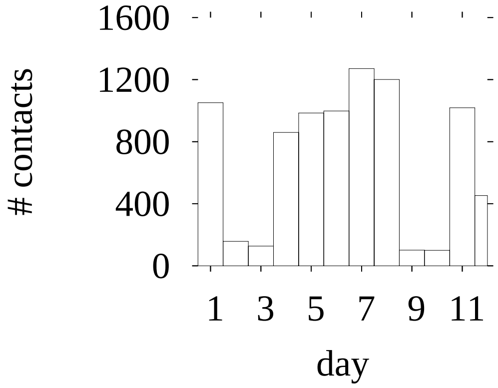


Figure 5.4: Number of contacts per day between mobile iMotes.

In 5.5 we can see the distribution of the inter-contact time between students. The inter-contact time is the time between two contacts for a given node pair, and its distribution has previously been shown to exhibit a power-law behavior in a large number of experiments[50]. We see similar tendencies to power-law behavior as in previous experiments here, but we can see that a large part (over 90%) of the inter-contact times are shorter than one day. This means that after a node pair have met, there is a 90% chance they will meet again within one day. The exponent of the power law is equal to 0.46.

5.3.2 Contacts with fixed iMotes

As explained previously, one of the goals of the experiment we present in this chapter was to explore not only the interaction between the participants wearing iMotes but also to capture their mobility from fixed locations distributed at popular places in the city. However, the results we obtain do not meet our expectations as shown by the plots in 5.6. 5.6(a) and 5.6(b) present, for each of the fixed iMotes, the total number of contacts with mobile iMotes and the number of unique mobile iMotes observed, respectively. They show in detail that very few contacts occurred between iMotes carried by students and most of the fixed ones. The only two fixed iMotes having significantly more contacts with students were those at the reception at the Computer Lab (where the students attend class activities) and at a popular grocery store.

There are a number of factors that can explain this result. First, it might be possible that the fixed iMotes were deployed at inappropriate locations according to the population sample.

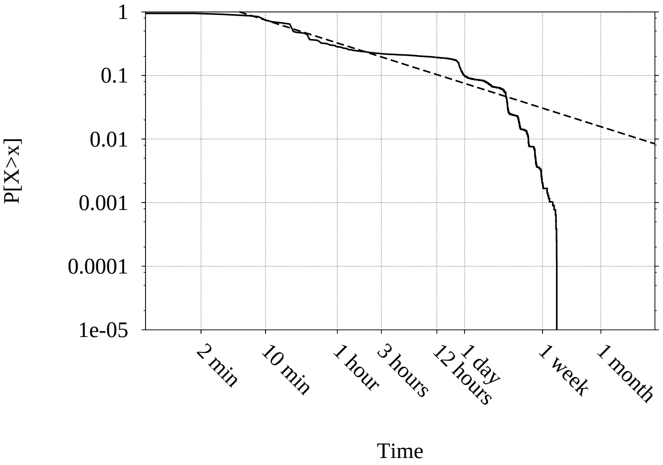
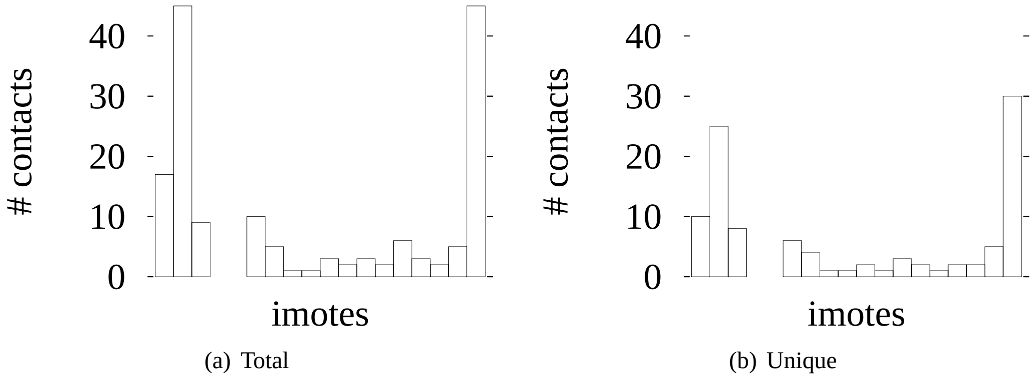


Figure 5.5: Inter-contact time distribution



(a) Total

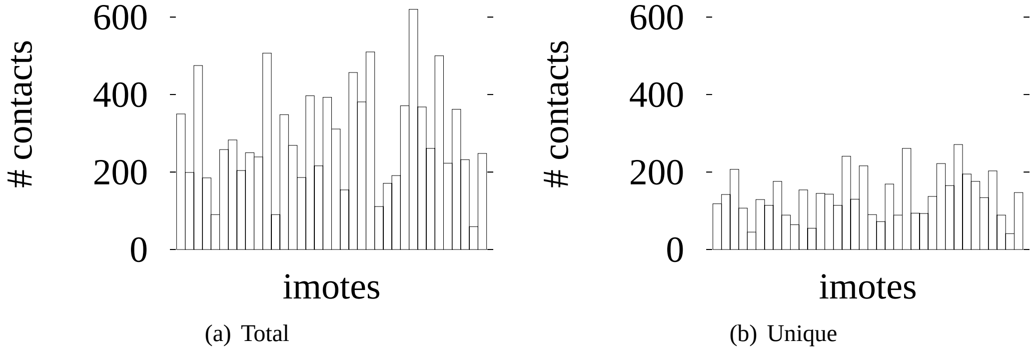
(b) Unique

Figure 5.6: Contacts between mobile and fixed iMotes.

Before the deployment, an attempt was made to survey students about popular locations to visit, and this in conjunction with reasoning on where people are likely to go (which is possible in a city of Cambridge's size), the locations were chosen. Apparently, students did not experience a large number of contacts with locations where we had deployed the iMotes. As we will see in the next section, the fixed iMotes did however log many external contacts, verifying that the locations in which they were deployed were indeed frequently visited by people with Bluetooth enable devices – just not experiment participants. This kind of deployment might work better in corporate environments in which people are confined all the day or in experiments with more participants. Second, we might have missed logging many contacts, especially in transit areas. This issue is discussed later in 5.3.4.

5.3.3 External contacts

In addition to measuring contacts between iMotes, all contacts between the iMotes and other Bluetooth enabled external devices were also logged. While this was not the main objective of the experiment, this data ended up constituting the largest part of our data set. Indeed, we observed $10{,}469$ contacts ($3{,}586$ unique) between mobile iMotes and external devices, and $20{,}240$ contacts ($9{,}211$ unique) between fixed iMotes and external devices. Here we investigate these contacts with external devices by first quantifying them and then trying to identify the nature of these devices.



(a) Total (b) Unique

Figure 5.7: Contacts between external devices and mobile iMotes.

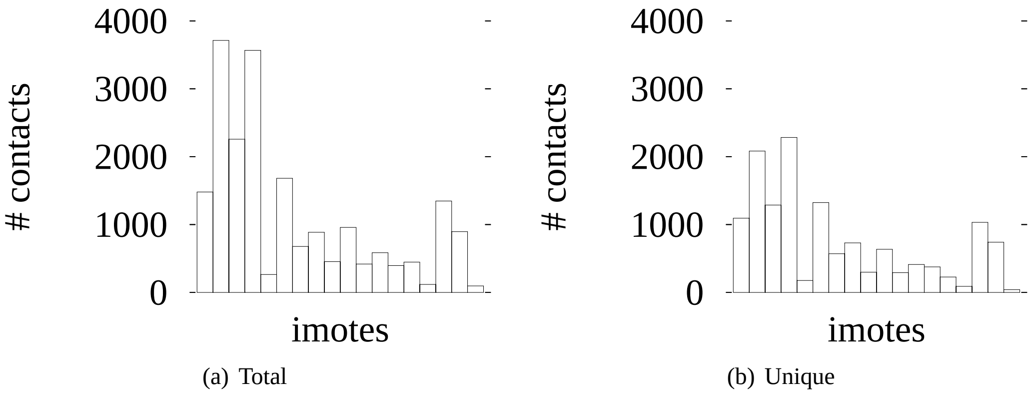
5.7 shows the contacts each mobile iMote had with external devices. 5.7(a) and 5.7(b) show the total number of external contacts and the number of unique external contacts respectively. Mobile iMotes acquired on average 290.8 external contacts and 139.9 unique external contacts with respectively a standard deviation of 132.6 and 139.9.

5.8 is similar to 5.7 but for the fixed iMotes. In that case, fixed iMotes acquired on average 1124.7 external contacts with a standard deviation of 1049.7 and 760.5 unique external contacts with a standard deviation of 632.3. The fact that the number of contacts is higher for the 4 first days is due to an iMote that ran rapidly out of memory, being placed in a very popular location. Note that we did not consider this iMote in simulations presented next section.

To continue our investigation on external devices, we used the database of Organizationally Unique Identifiers (OUI)[2] maintained by the IEEE to map MAC address prefixes in the data set to their manufacturers. We were able to resolve 97% of the prefixes. 5.9 presents the frequency of the occurrence of the most common manufacturers. From looking at the manufacturers, we can see that most of external devices are likely mobile phones or other portable devices (Murata

---

[2] http://standards.ieee.org/regauth/oui/



(a) Total (b) Unique

Figure 5.8: Contacts between external devices and fixed iMotes.

is a Bluetooth chip manufacturer whose products are integrated in a wide range of devices such as mobile phones, Personal Digital Assistants (PDAs), laptops, etc.), as opposed to devices such as printers or wireless keyboards and mice. This observation is of great interest because we can reasonably consider external devices in the data set as mobile entities that are carried in pockets of regular people moving around the city, exactly as mobile iMotes.

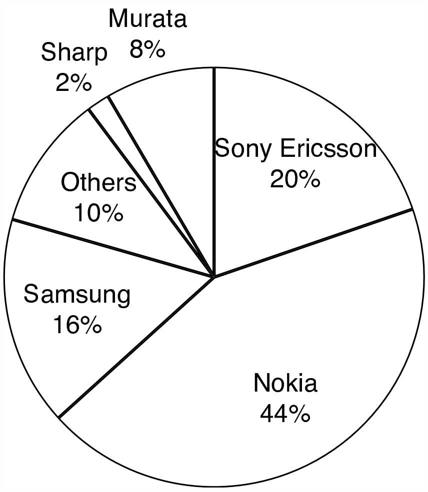


Figure 5.9: Manufacturers

5.3.4 Discussion

When using devices like the iMotes to gather data about contact patterns, measurements may not be able to exhaustively capture contacts because of two main factors. First, as these experiments require the active involvement of participants, there is always the risk that the participants do not completely fulfill their commitments, for example by not always carrying the measurement devices everywhere they go. Indeed, as we conducted a survey on our population after the experiment, we were able to determine that some of them had occasionally forgotten to bring the iMote when going somewhere, or might have left it in a bag instead of keeping it on their person. There were also occasions of students leaving the city over the weekend (which is less of a problem as that reflects a real user behavior, and will still be able to gather external contacts, but most likely no internal contacts). Secondly, as contacts are only discovered using the periodic Bluetooth inquiries, iMote experiments suffer from a sampling effect that means that contacts that are shorter than δ minutes may be missed. This is a trade-off between minimizing the risk of missing short contacts and the life-time of the iMotes. It would be possible to extend the life-time further while keeping a short inquiry interval by adding more powerful batteries, but that would result in a more bulky form factor.

5.4 City-wide content distribution

5.4.1 Scenario

We propose here an evaluation of a city-wide content distribution architecture. As said previously, this architecture is composed of wireless short range access points disseminated down town at popular places. Content (e.g., newspapers or local information) is opportunistically distributed to nodes that pass close to these access points. We consider fixed iMotes we deployed in Cambridge to be the content distribution access points and mobile iMotes given to students being the target group. Our aim here is to propose and evaluate schemes that distribute content to a population of users interested to which our target group is assumed to belong (i.e., students from Cambridge University wearing mobile iMotes). Members of the target group were not aware of the positions of access points.

In this scenario, access points generate a new copy of a given newsaper at 7 am every day. Once acquired, copies are kept by nodes till 7 am the next day. We have replayed in this evaluation 5 days of data gathered in Cambridge from Monday to Friday. We removed the fixed iMote at the Computer Lab because it was located at the place the community of student met most of the time (we did not want to reduce the study to a trivial exercise).

5.4.2 Distribution schemes

Within the scenario previously described, we evaluated the following distribution schemes:

- *Selfish*: nodes get the content directly from the access points and never pass it on to other nodes. The access points distribute an unlimited number of copies.
- *Collectivist*: nodes can get content directly from access points and are able to share it within the communities they belong to. Note that nodes in our target group are assumed to belong to the same community.
- *Extended collaboration*: in addition to *Collectivist* strategy, external mobile devices can be used to relay the content. The details of the schemes we propose are presented later in this section. External mobile devices may be of several kinds. In the case of a collaborative scheme may involve strangers, mechanisms to incite nodes to relay the content need to be provided but are not the focus of this chapter. Also note that the experiment we conducted did not provided us contacts between external devices.
- *Top students*: use only the N students that had the highest number of contacts to be able to pass copies to the others.
- *Strangers only*: students can not relay the content, only external devices are used as relays.

In order to define heuristics to select the external devices that would be involved in schemes using extended collaborations, we obtained statistics on potential *mobile bridges*, defined as nodes having seen at least a fixed and a mobile iMote during the experiment. These nodes represent 12.5% (1,430 over 11,367) of external devices, they are potentially interesting to act as relays between access points and the targeted population.

In 5.10, we plot one point for each of the mobile bridges, showing the number of contacts it had with fixed and mobile iMotes. We add some small random noise in order to obtain a cloud of points. We can first observe that no mobile bridges had a large number of contacts with both fixed and mobile iMotes. A given mobile bridge seems to be close in terms of its mobility to either a fixed iMote or to a mobile one. Mobile bridges had on average 3.8 contacts with fixed iMotes with a standard deviation of 5.6 and 4.3 with mobile iMotes with a standard deviation of 19.3.

We define a pair here as a set of one fixed and one mobile iMotes that could be potentially covered by a mobile bridge, meaning that there is at least one mobile bridge that has seen these mobile and fixed iMotes. We found that 610 pairs between the sets of fixed and mobile iMotes

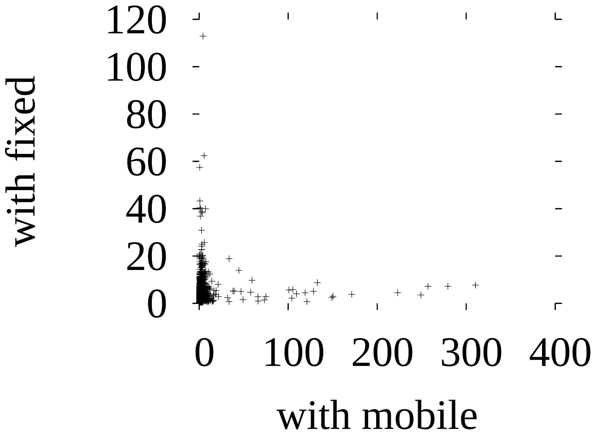


Figure 5.10: Number of contacts with fixed and mobile iMotes.

exist in the data over the 680 possible pairs. To have a better idea of the strength of the coverage of pairs by mobile bridges, we plot in 5.11 the probability distribution that pairs are covered by more than X mobile bridges. This figure shows that some pairs are covered by a significant number of mobile bridges. 10% of the pairs are covered by more than 20 bridges.

All these observations concerning mobile bridges lead us to think that strong hidden connections exist between the two sets that as first seen to be largely disconnected (as seen in 5.3.3, the number of contacts between fixed and mobile iMotes have been observed to be low). They also motivate the definition the following variations for the extended collaboration scheme:

- *All external*: use all external devices as relays.

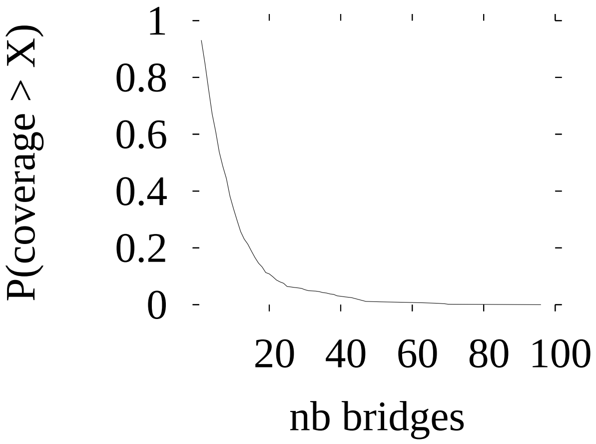


Figure 5.11: Coverage of pairs by mobile bridges.

- *Top external*: use the N external devices that had the highest number of contacts.
- *Top coverage pairs*: use the N mobile bridges covering the highest number of pairs (fixed, mobile).
- *Top coverage contacts*: use the N mobile bridges having the highest number of contacts with both mobile and fixed iMotes. Note that for all the 3 previous schemes, an algorithm have to be defined to choose these high potential relays in a distributed fashion. We let this for further work.

5.4.3 Performance evaluation

To measure the performance of the different content distribution schemes, we used the following metrics:

- *Delivery ratio*: the percentage of *bundles* (from the DTN terminology, i.e., messages containing the electronic newspaper) that were delivered. In our scenario, the maximum number of bundles that could be delivered is 175 (1 bundle is expected for each of the 35 mobile nodes each of the 5 days of simulation). This metric evaluates the user satisfaction.
- *Average delay*: the average bundle delay (computed on the bundles delivered).
- *Efficiency*: the number of messages transmitted per bundle delivered. It represents a measure of the network resource usage.

5.2 presents the simulation results. The first thing that we can observe is that the selfish strategy leads to poor results in delivery ratio (20.5%), which seems natural since we did not measure a large number of contacts between students and access points. However, we see a great improvement when the students collaborate, leading to 90.2% of delivery. Moreover, what we can see from the results regarding the extended collaboration scheme is that delivery ratio is slightly improved when increasing the number of relays selected while the delay is significantly decreased being close to the minimum that can be achieve with our data (i.e., 4.10 hours when using all the nodes). Note that the top coverage contacts or the top coverage pairs seem to be the most efficient strategies among the ones evaluated when selecting a small number of external devices as relay. With N=10, we reduce the delay for the top coverage contacts by 20% while increasing the delivery ratio from 1.4% compared to the collectivist scheme. Finally, external devices seem not to be sufficient to ensure a high delivery ratio themselves. When only using strangers to relay the content we only achieve a 66.2% delivery ratio.

	Delivery	**Delay**	**Efficiency**
Selfish	20.5	7.47	1.00
Collectivist	90.2	5.29	1.00
All external	97.1	4.10	36.4
Top external			
N=1	90.2	5.29	1.03
5	90.2	5.23	1.15
10	91.4	4.45	1.28
50	92.5	4.50	2.26
100	94.2	4.60	3.33
Top cov. pairs			
N=1	90.2	5.29	1.02
5	91.4	4.44	1.13
10	91.4	4.44	1.26
50	94.2	4.60	2.09
100	95.4	4.59	2.86
Top cov. contacts			
N=1	90.2	5.29	1.03
5	90.2	5.25	1.11
10	91.4	4.40	1.21
50	93.7	4.47	2.00
100	94.8	4.45	2.78
Top students			
N=1	20.5	7.46	1.00
5	56.5	9.52	1.00
10	66.2	7.55	1.00
35	90.2	5.29	1.00
Strangers only	66.2	8.06	40.99

Table 5.2: Simulation results.

What the results tell here is that, collaboration inside the target group improves a lot the performance while only adding one transmission per bundle delivered compared to the selfish strategy. Furthermore, a slight gain in delay and a significant one in delivery can be achieved when using a few nodes that are *close* both to students and to access points.

5.4.3.1 Robustness with number of access points

We evaluate here the *robustness* of this content distribution infrastructure by looking at its performance if some of the most popular access points are removed (the ones with highest numbers of contacts). We removed access points in order from the most popular one to the least. 5.12 presents the results in delivery ratio for the following schemes: collectivist, all external and extended collaborations (top coverage pairs and top coverage contacts with N=10). It shows that the interest of using external devices as relays is clear when the number of access points is decreased. Extended collaborations achieve a delivery ratio in between collectivist and all external.

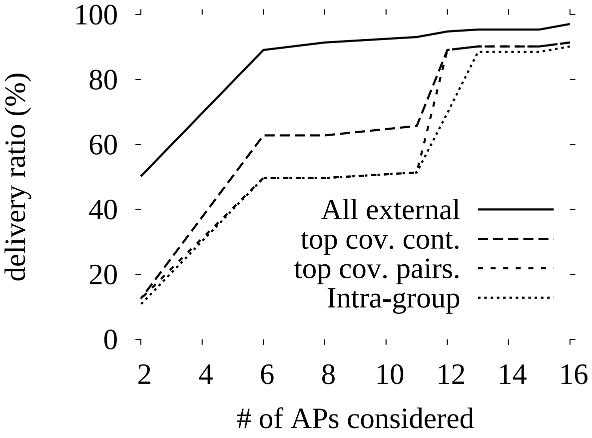


Figure 5.12: Delivery ratio when removing popular APs.

5.4.3.2 Robustness with number of students

Similarly, we evaluate here the *robustness* of this content distribution infrastructure by looking at its performance if most popular students are removed (the one with highest number of contacts). We removed students from the most popular to the less one. 5.13 presents the results in delivery ratio for the same schemes as previously. Again, we see clearly that the use of external devices increases the delivery ratio when the number of members of the community is decreased.

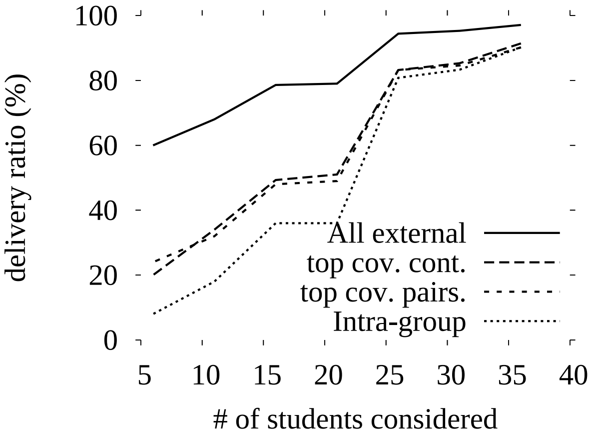


Figure 5.13: Delivery ratio when removing popular students.

	Intel	**Cam-U**	**Infocom05**	**Cambridge**
Duration (days)	3	5	3	10
δ (mins)	2	2	2	10
Devices	8	12	41	36
Internal contacts	1091	4229	22459	8545
Average # Contacts/pair/day	6.5	6.4	4.6	1.5
External devices	92	159	197	3586
External contacts	1173	2507	5791	10469

Table 5.3: Comparison with previous experiments for data from mobile iMotes.

5.5 Related work

Efforts to acquire mobility data for DTN scenarios have expanded rapidly in the past couple of years. The Reality Mining [90] experiment conducted at MIT has captured proximity, location, and activity information from 100 subjects over an academic year. Each participant had an application running on their mobile phone to record proximity with others through periodic Bluetooth scans and location using information provided by the phone on the cellular network. The UMass DieselNet project [53] also aims to study DTN routing in challenging contexts such as power outages or natural disasters. A testbed to gather interactions between 40 buses in western Massachusetts was deployed in 2005. In addition to the experiment described in this chapter, other experiments with iMotes have been conducted by the Haggle [50] project, which explores networking possibilities for mobile users using peer-to-peer connectivity between them in addition to existing infrastructures. To show the similarities and differences between this and previous studies, we summarize the main parameters and measurement results from all the experiments in 5.3, extending the information provided by Chaintreau et al. [50].

The experiments *Intel* and *Cam-U* were performed in corporate and research lab settings, with the participants being researchers and graduate students. The *Infocom05* experiment was conducted at a research conference and the *Cambridge* experiment is the experiment presented in this chapter. We see that this experiment spans 2-3 times as much time as previous experiments with similar number of mobile devices as the conference experiment, but with significantly more iMotes than in the first two experiments. In the other experiments, we see a high number of internal contacts while in this experiment, whereas in this experiment this number is much lower. On the other hand, the number of external devices seen and the number of contacts with them are much higher in this experiment than in previous ones. Both of these differences can be explained by the population of participants and the setting in which the experiment was deployed. In the previous experiments, participants were chosen from either people that work together on a daily basis at the same premises or attend the same conference. Thus, it is natural that they will have frequent contacts with each other. On the other hand, in this experiment, students might not have pre-existing relationships with each other and are thus less likely to have contacts outside class activities.

Other work has been to gather data that can be used, after some processing, as DTN-like data. For instance, Henderson et al. at Dartmouth College [48] have deployed one of the most extensive trace collection efforts to gather information about its Wi-Fi access network. These data have been used as mobility data to characterise the mobility of users [49] or to evaluate DTN routing protocols [10]. Similar Wi-Fi based data have been used to analyse mobility such as that of ETH Zürich [51]. Furthermore, the data presented in this chapter might be of great interest to evaluate forwarding algorithms defined for DTNs such as the work by Vahdat et al. [25] that uses epidemic routing, the Spray and Wait [29] protocol that distributes a number of copies to relays and then waits until the destination meets one of them, MobySpace [10] that uses a virtual space based upon nodes' mobility patterns, or the PRoPHET routing protocol[33], which bases routing on a probabilistic metric calculated using history of encounters and transitivity.

Finally, a very close scenario to that of our work was introduced in a research note by Lawrence et. al[94]. They envision to create a community content distribution network using *familiar strangers* [95], i.e. people who we meet very regularly but who we do not know. This kind of node may be present in the data we collected.

5.6 Conclusion

We have proposed and evaluated in this chapter schemes for distribution of content in a urban environment using short range Bluetooth access points. To evaluate these schemes, we conducted a city-wide experiment using Intel iMotes, which are Bluetooth contact loggers. Stationary iMotes were deployed at popular places to act as content distribution access points while students, considered as our target group, from Cambridge University were carrying other iMotes in their pockets. We show that the simple fact that students collaborate led, in this experiment, to a delivery ratio of 90% and that the additional use of Bluetooth devices external to the experiment to relay the information slight increased the delivery ratio while significantly decreasing the delay. We have also shown that the interest of using external devices as relays increases when the size of the infrastructure and of the targeted communities decreases. Finally, we introduced here a new kind of data set and make it available to research community.

Acknowledgements

We would like to thank all the participants of the experiments and we especially acknowledge Pan Hui for his useful support. He helped us to program and package the iMotes, and to extract data from them after the experiment.

Chapter 6

Conclusion

Delay tolerant networking has emerged in the past couple of years as a possible means for extending the current Internet architecture to support challenged networks. These networks are mainly characterized by the fact that connectivity between entities suffers from disruptions. From the link perspective, links could exist only intermittently, have highly asymmetric data rates, present large propagation delays or suffer from high error rates. At network scale, the heterogeneity of links and of communication stacks lead Internet-like networking solutions to fail. DTNs are conceived to address these issues.

This thesis represents our contributions to DTN routing. We addressed, in particular, the following issues:

- *Characterization of contact patterns*: Understanding how entities interact with each other is of great interest for DTN routing. In the context we mainly address in this thesis, node mobility and interactions are driven by social behaviors. We tried to answer the following questions: How nodes interact with each other? Do the interactions between nodes follow well known distributions? Is there heterogeneity in node interactions?

- *Knowledge-based routing and content distribution*: Routing in DTNs is more efficient when nodes can use knowledge about network connectivity to make routing decisions. We addressed the following questions: What kinds of knowledge provide the most efficient routing? Is this knowledge predictable from past information? How can this knowledge to be exploited? It is possible to mix replication based approaches with knowledge based ones to improve performance?

- *Collection of contact patterns*: Mobility models for DTNs are still in their early stages. Collecting interactions between entities in real world contexts is of interest to the community. However, few traces are available publicly. Data such as those we have provided can not only be used to validate or define mobility models, but can also serve as an input to simulation work.

We addressed these issues by conducting studies that present results of interest for the networking community. We were one of the first to focus on real connectivity data in terms of: analyses that help our understanding of contact patterns between nodes, evaluations of routing schemes by replaying data in simulations, and a data collection effort with the conduction of an iMote based experiment involving both mobile proximity sensors carried by students and fixed ones placed at popular locations at the scale of a city.

In Sec. 6.1, we describe what we accomplished and the results we obtained. In Sec. 6.2, we present directions for future work to continue and extend the research described here.

6.1 Contributions

In this thesis, we made several contributions to delay tolerant networking in scenarios where network entities are mobile (e.g., mobile phones, PDAs) and carried by people who have overlapping social relationships.

First, we showed, with the analysis of real traces, that there is heterogeneity in interactions between participants in such networks and we demonstrated that this heterogeneity could be exploited for developing efficient routing schemes. Second, to move forward in the direction suggested by our first contribution, we proposed single-copy and multi-copy routing algorithms based on the use of a high-dimensional Euclidean space, that we call MobySpace, constructed upon knowledge about nodes' mobility or connectivity patterns. We have shown, through the replay of real mobility traces, that MobySpace-based routing schemes can applied to DTNs and that this can bring benefits in terms of enhanced bundle delivery and reduced communication costs. Finally, to contribute to the on-going data collection effort, we presented an analysis of contact traces that we collected in an experiment we conducted in Cambridge, UK. The aim of this experiment was to study the feasibility of a city-wide content distribution architecture composed of short range wireless access points.

The following are extended summaries of our contributions to the DTN domain:

First, in Chapter 3, we argue for the wisdom of using pairwise inter-contact patterns to characterize DTNs. We have first provided a statistical study using widely-used DTN data sets in which we characterize heterogeneity of interactions between nodes. We show that pairwise inter-contact times processes, which have a great impact on routing, are heterogeneous and distributed in log-normal for a large number of node pairs. Second, we describe the power-law paradox and show that the distribution of aggregate inter-contact times can be power-law distributed while pairwise processes are exponentially distributed. Finally, we have validated the insight that taking heterogeneity into account for routing improves performance. We presented a new routing strategy, SW*, which is capable of using only a subset of relays to improve routing performance, measured in terms of average delay. We show, by replaying real connectivity traces, that SW* achieves good performance,in terms of delivery ratio and delay, while keeping the overhead low. We also discussed factors and implementation issues that might have impacted the results.

In Chapter 4, we have proposed and validated of a generic routing scheme that uses the formalism of a high-dimensional Euclidean space constructed upon mobility patterns, the MobySpace. We have shown through the replay of real mobility traces that it can be applied to DTNs and that it can bring benefits in terms of enhanced bundle delivery and reduced communication costs. We have also presented results of a feasibility study in order to determine the impact of the characteristics of nodes' mobility patterns on the performance and to study nodes' ability to learn their patterns. To make DTN routing work with MobySpace, nodes need to have a minimum level of mobility with mobility patterns that can be sufficiently discriminated. We present encouraging results about the capacity of nodes to learn their own patterns. We also see that nodes can reduce the number of components of the mobility patterns they store without great impact on routing performance. This can reduce the overhead of MobySpace and the complexity of handling mobility patterns.

Finally, in Chapter 5, we have proposed and evaluated schemes for distribution of content in a urban environment using short range Bluetooth access points. To evaluate these schemes, we conducted a city-wide experiment using Intel iMotes, which are Bluetooth contact loggers. Stationary iMotes were deployed at popular places to act as content distribution access points while students, from Cambridge University, considered as our target group, were carrying other iMotes in their pockets. We show that the simple fact that students collaborate led, in this experiment, to a delivery ratio of 90% and that the additional use of Bluetooth devices external to the experiment to relay the information slightly increased the delivery ratio while significantly decreasing the delay. We have also shown that the interest of using external devices as relays increases when the size of the infrastructure and of the targeted communities decreases.

Finally, we introduced a new kind of data set and made it available to the research community via the CRAWDAD [45] data repository.

6.2 Perspectives

In this section, we present some of the research perspectives which we envision as a follow up to the thesis. Sec. 6.2.1 describes future work to extend our contributions and Sec. 6.2.2 points out research issues, not directly connected to our contributions but that we consider as important, for future work on delay tolerant networking.

6.2.1 From our contributions

We describe here future work that could extend the contributions of this thesis.

Clearly, our work presented in Chapter 3 on the characterization and use of the heterogeneity of inter-contact time distributions, will benefit from studies of correlations between processes, and of short and long term dependencies in DTN data sets. Furthermore, formal studies should be conducted of more elaborate schemes, in terms of number of copies distributed or in terms of the number of hops traversed. Finally, more realistic evaluations would need to take into account limitations on buffers and bandwidth.

Future work that might follow our work on MobySpace in Chapter 4 includes studies concerning the impact of the structure of the Euclidean space, i.e., the number and type of dimensions, and the similarity function. Different kinds of Euclidean space can be investigated by considering schemes like the one described in Sec. 4.3 that takes, for each dimension, the frequency of contacts between a certain pair of nodes or the one that captures cyclic frequential properties during nodes' visits to locations. Work remains to be done on the stability of mobility patterns over time and their ability to be learned by nodes. The patterns may contain long term and short term dependencies, as pointed out by Ghosh et al. [96]. Nodes can have different mobility patterns that are each stable. For instance, they can have one for the weekends, one for the vacations, and one for working weeks. Additionally, further validation need to be conducted using real data and in different environments. MobySpace can be tested on traces coming from larger cell networks, like GSM networks. We might also want to evaluate MobySpace in different social contexts where nodes have specific mobility patterns.

Finally, future work which might follow our contribution to content distribution in Chapter 5 includes studies that use these data as an input to propose DTN mobility models, producing interactions between entities similar to the one observed in this chapter. Also, these data, in addition to others available from CRAWDAD [45], can be used to study communities of people. Having the knowledge of such communities or being able to detect them would be of great help to propose efficient communication schemes. Finally, these data can be used as an input to simulators to evaluate protocols designed for DTN scenarios.

6.2.2 On related issues

Beside research activities envisioned as a follow up to the contributions of this thesis, our experience in the domain has led us to a certain number of issues that we want to address in the future. We list some of them here:

- *Traffic patterns*: Traffic may not take a unicast form in most PSN or DTN deployments. People might have group communication needs instead of unicast ones in which the level of delay and loss tolerance seem to be lower. Work needs to be conducted to see if our contributions can be extended to achieve such communications.

- *Security issues*: Peer-to-peer communications using opportunistic wireless connections between mobile entities is very exiting and worrying at the same time. Indeed, we can easily envision that people can exchange information without any control for illegal or immoral purposes and it would be especially difficult to police. Furthermore, allowing people to communicate and relay information through other people devices raises a number of security issues: computer virus, intrusions, etc. Viruses appeared a couple of years ago on smart phones [1], They automatically send themselves from one phone to another via the Bluetooth facility.

- *Back to MANETs*: Our interest for delay tolerant networking was mainly guided by the fact that MANETs suffer from severe connectivity issues, which is a barrier to their deployment and use in real scenarios. We then headed toward extreme networks in which the level of connectivity is very low. Our work in this domain leads us to think now that we need to propagate DTN ideas and research results back to MANETs, to increase their tolerance to connectivity disruptions and to improve their overall performance. Already in this direction, Ott et al. [97] have worked on the integration of DTN and MANET routing. If connectivity is sufficient, ad hoc routing is performed using AODV. Otherwise, still using AODV, DTN gateways in the neighborhood are discovered and used. There is potential for more developments along these lines.

- *Resource management*: We did not provide any mechanism to route around congestion or avoid message drops if storage is unavailable or the battery level of a node is too low. Our contributions were mainly to validate factors that routing schemes may integrate,

---

[1] `http://www.computerworld.com/securitytopics/security/story/0,10801,97935,00.html`

such as the heterogeneity of contact patterns. However, in a real deployment, resource management is a critical issue that has to be taken into account in the design of each system component. We would like to conduct studies in which resource management is at the center of every operation.

- *Connectivity opportunism*: As pointed out in the PSN scenario, we are more and more in presence of different infrastructures while moving in our environment. Taking that into account in studies around delay tolerant networking might drastically change some of the results that we have obtained and impact the way networking technologies are designed. Lindgren et al. [98] have tried to quantify the way in which the presence of infrastructure would change the shape of the inter-contact time distributions that were introduced by Chaintreau et al. [47].
- *Environment characterization*: Work on the description of environments in which DTNs might be deployed also needs to be done. It may range from scenario definition to new trace collection efforts. Understanding the underlying mechanisms that drive entity interactions should lead to the definition of better mobility models to be used for the validation of DTN networking schemes.

Appendix A

La Problématique

De nos jours, les technologies de communications sont au coeur des systèmes informatiques. Ces systèmes se composent d'entités intégrant un nombre grandissant de technologies de communication et ayant de plus en plus besoin d'échanger de l'information entre eux. Les environnements dans lesquels ces systèmes opèrent peuvent avoir des contraintes et des méthodes de communication spécifiques. Les entités étant de plus en plus mobiles (téléphones mobiles, PDA, ordinateurs portables), la connectivité aux réseaux devient intermittente, les opportunités de communication pouvant avoir des durées très variables. Par ailleurs, les entités mobiles, en portée radio par intermittence, devraient être capables de s'échanger de l'information soit pour palier à un manque temporaire d'infrastructure, soit pour supporter des modes de communication dit ad hoc. Ces réseaux ad hoc pouvant être déconnectés de manière très forte (e.g., un réseau urbain de bus utilisant la technologie Wi-Fi pour s'échanger de manière opportuniste des informations de trafic). L'hétérogénéité des systèmes actuels et leurs nouveaux besoins en communication ont fait naître des mécanismes réseaux permettant d'étendre l'architecture actuelle de l'Internet dans un contexte où la connectivité est très hétérogène et intermittente. On regroupe l'ensemble des travaux et résultats permettant le support des communications dans ces environnements sous l'appellation de Réseaux Tolérants aux Perturbations ou aux Délais (en anglais, Disruptive/Delay Tolerant Networks (DTN)).

Les travaux autour des DTNs sont issus de recherches sur les réseaux interplanétaires (en anglais, Inter-Planetary Network (IPN)), menés au sein du groupe de travail IPNSIG (IPN Special Interest Group) depuis 1998. Le groupe DTNRG [13] (Delay Tolerant Network Research Group) a été créé en 2002 à l'IRTF (Internet Research Task Force) afin de recentrer les efforts sur la définition d'une architecture et de protocoles pour les DTNs.

Le groupe DTNRG étudie des réseaux dits *difficiles* (en anglais, *challenged networks*) ayant les propriétés suivantes:

- *Connectivité intermittente*: La connectivité souffre de problèmes liés à des turbulences dues par exemple à la mobilité des noeuds, aux variations dans la propagation du signal radio ou à la gestion de l'énergie des noeuds.

- *Délais de propagation*: Les liens du réseaux peuvent avoir de long délais de propagation si bien que les protocoles usuels de l'Internet tel que TCP sont incapables de transporter de l'information. Ces liens peuvent être des liens satellites, des liens sous-marins ou des liens assurés par des mules (entités transportant physiquement des données d'un point de rencontre à un autre).

- *Débits asymétriques*: Les débits sur les liens sont fortement asymétriques. Ils peuvent être, dans le pire des cas, unidirectionnels.

- *Taux d'erreur important*: Les réseaux peuvent contenir des liens où le taux d'erreur est très grand, créant des goulots d'étranglement.

Dans cette thèse, nous abordons uniquement les réseaux DTNs composés d'entités mobiles ayant des capacités de communication sans-fil. Ces entités sont connectées entre elles par intermittence principalement en fonction des interactions sociales qui lient les personnes qui les transportent (e.g., étudiants équipés de téléphones Bluetooth se déplaçant sur un campus, étant ou pas dans le même programme).

La connectivité est un problème majeur dans les réseaux composés d'entités fixes ou mobiles. Bien comprendre la connectivité d'un réseau permet d'adapter les mécanismes de communication afin d'assurer le bon fonctionnement des applications. La non prise en compte de ce facteur peut engendrer l'impossibilité totale pour les entités de communiquer. Les contacts entre les entités peuvent, de manière générale, être classés en trois catégories [5]:

- *Les contacts déterministes*: Ces contacts sont programmés et peuvent donc être anticipés de manière précise. Ils ont lieu par exemple dans le cas de veilles et réveils périodiques de capteurs.

- *Les contacts prévisibles*: Il s'agit de contacts pouvant être prédits avec l'utilisation d'historiques. Par exemple, si un noeud a été en contact avec le noeud A quotidiennement ces dix derniers jours, il y a une grande probabilité pour que cela se reproduise aujourd'hui.

- *Les contacts opportunistes*: Ces contacts ne sont pas prévisibles, ils s'établissent de fait.

Le relayage des paquets de données ne pouvant se faire avec les mécanismes de routage IP classiques (e.g., Internet) dans les réseaux DTNs, il se base alors sur le principe suivant : stockage et transmission (en anglais, Store and Forward). Les noeuds DTN ont en effet une mémoire tampon dans laquelle ils peuvent stocker temporairement des messages. Lorsqu'une entité reçoit un message à retransmettre, elle le conserve jusqu'à ce qu'elle rencontre le destinataire ou le transmet de manière opportuniste à un relais. L'architecture DTNRG consiste en la définition d'une couche de transport appelée bundle layer. Cette couche permet d'acheminer des messages, appelés bundles, dans le réseau. Les bundles sont transférés atomiquement entre les noeuds de proche en proche, en utilisant TCP par exemple à chaque saut. Les bundles ne sont pas limités en taille et les noeuds dans cette architecture doivent avoir des mémoires tampon assez grandes afin d'assurer le stockage temporaire des bundles.

Dans les DTNs, l'architecture et les mécanismes réseaux n'ont cependant pas encore atteint la maturité, un certain nombre de problèmes doivent être approfondis, tels que:

- *Transport*: Les questions suivantes doivent être abordées: quels protocoles de transport doit on utiliser dans les DTN? Quels mécanismes d'acquittements, de contrôle de flux et de congestion doit utiliser le transport?

- *Management des ressources*: Les appareils impliqués dans les DTNs ont des capacités limitées en termes de puissance de calcul, de batteries et de stockage, il faut donc définir des stratégies intégrant un management fin des ressources disponibles.

- *Adressage*: L'utilisation opportuniste de plusieurs interfaces par les appareils ainsi que la nature intermittente de la connectivité font de l'adressage un challenge. Des solutions robustes aux déconnections doivent être mises en place.

- *Routage*: Le routage est également un challenge dans la mesure où la connectivité est intermittente. Des solutions intégrant des éléments de contexte et des connaissances de la connectivité du réseaux doivent être proposées.

- *Intéropérabilité*: Les solutions DTN doivent pouvoir: (1) supporter les protocoles utilisés dans l'Internet ainsi que (2) pouvoir éventuellement assurer les communications DTN de manière transparente pour les applications déjà existantes.

Dans cette thèse nous nous sommes principalement intéressé à la problématique du routage. Les protocoles du monde ad hoc comme AODV [99] ou OLSR [100] ne pouvant pas fonctionner, d'autres approches doivent donc être considérées. Les solutions existantes pour le routage peuvent se classer en trois catégories:

- *Utilisant la réplication*: Dans les DTNs, l'amélioration des performances pour le routage passe souvent par l'ajout de diversité dans les opportunités de contact saisies, c'est pourquoi une large partie des protocoles utilise la réplication. Dans le routage épidémique [25], les messages se propagent de manière virale dans le réseau. A chaque noeud rencontré, une synchronisation s'effectue afin de récupérer les messages non précédemment acquis. Il s'agit de la solution la plus efficace en terme de délais mais la plus consommatrice en ressources radio et en mémoire tampon sur les noeuds. Dans le cas d'un routage curatif [30], une fois qu'une phase de routage épidémique a permis d'acheminer le message vers la destination, celle-ci envoie également un message de manière épidémique qui aura pour but de nettoyer les mémoires tampons des noeuds afin que le message original ne soit plus propagé et éliminé. Enfin, pour le routage de type Spray [29], à la réception d'un nouveau message, un noeud décide de retransmettre N fois ce message aux noeuds qu'il rencontrera. Il existe beaucoup de variantes pour le choix du nombre N et également pour le choix des relais à qui seront transmises les copies [30].

- *Utilisant des connaissances*: Les méthodes de cette catégorie traitent uniquement le fait de considérer les contacts comme des opportunités ou non. Si un noeud juge que le noeud qu'il rencontre n'a strictement aucune chance de l'aider pour la livraison d'un message, il peut décider de ne pas lui transférer. Une série de travaux ont été mené par Jain et al. avec des contacts déterministes [23], par exemple dans le cadre d'un rattachement à Internet d'un village en Afrique via trois moyens de communications: un satellite basse orbite, une moto et un modem RTC. Akyildiz et al. [7] ont également étudiés des algorithmes de routage pour des réseaux interplanétaires. Concernant les contacts prévisibles, Lindgren et al. [33] ont proposés un algorithme se basant sur le fait que les entités sont organisées en communauté et que leurs déplacements sont liés à leur appartenance à telle ou telle communauté. Burns et al. [36] ont également introduit un algorithme se basant sur des informations d'historique sur les contacts entre les noeuds.

- *Approches hybrides*: Les deux approches précédentes sont intéressantes: la première utilise la diversité des noeuds pour augmenter les performances du routage et la seconde utilise une certaine connaissance de la connectivité pour atteindre le même but. Ces méthodes présentent néanmoins des inconvénients: (1) les méthodes basées sur la réplication n'utilisent pas de manière efficace les opportunités de contact, conduisant donc à un surcôut inutile, (2) les méthodes utilisant des connaissance sur la connectivité ne sont pas robustes aux longues déconnections ou à la disparition des noeuds car elles ne disséminent qu'une copie du message. Des approches hybrides sont donc nécessaires dans les DTNs. Peu de solutions hybrides ont été proposées pour le moment. Jain et al. [44] proposent d’utiliser des techniques de codage pour générer des blocs de données redondant, acheminés à la destination par des chemins différents.

Dans cette thèse, nous contribuons au domaine des DTNs par des propositions pour le routage innovantes pour le routage qui se classent dans les catégories ”avec connaissances” et ”approches hybrides”. Nous mettons également l’accent sur l’évaluation de ces protocoles en utilisant des traces de données réelles. Enfin, nous participons à l’effort de collection de traces d’interactions dans le cadre de l’étude d’un scénario de distribution de contenu à l’échelle d’une ville.

Appendix B

Contributions de cette thèse

Dans cette annexe, nous présentons les contributions de cette thèse. Cette discussion s'établie en trois phases. La partie B.1 présente une discussion autour de la nécessité de considérer les hétérogénéités dans les interactions qui existent entre les noeuds DTNs. La partie B.2 présente un formalisme, le MobySpace, permettant le routage de messages dans ces réseaux. Enfin, la partie B.3 consiste en l'étude d'un scénario de distribution de contenu que nous avons pu évaluer grâce à la conduite d'une expérimentation dans laquelle nous avons collecté des traces d'interactions réelles.

B.1 Importance pour le routage de l'hétérogénéité

Les travaux antérieurs sur le routage dans les DTN ont souvent fait l'hypothèse que les distributions des temps d'inter-contacts sont homogènes pour toutes les paires de noeuds du réseau. Le principal argument de notre contribution est que les travaux de recherches doivent prendre en compte le cas hétérogène. Chaintreau et al. [50] ont suggéré l'existence de cette hétérogénéité, ce que nous avons vérifié en analysant des traces réelles. En étudiant les données Wi-Fi de Dartmouth [48] ainsi que les contacts Bluetooth des projets Reality Mining [54] au MIT et IST Haggle [50], nous montrons qu'une très large proportion de paires ont des interactions distribuées en log-normale en terme de temps d'inter-contacts. De plus, nous démontrons que la distribution des temps inter-contacts agrégés mise en évidence dans [50] peut être obtenue par une composition de distributions exponentielles, pour chaque paires de noeuds, ayant des paramètres hétérogènes. Cette mise en évidence et le modèle pour les DTNs que nous proposons nous permettent d'étendre les travaux de Spyropoulos et al. [78, 79]. Nous montrons que les stratégies de routage peuvent bénéficier de la prise en compte de l'hétérogénéité des distributions des temps inter-contacts pour améliorer leur performance en terme de délai de livraison. Nous proposons un algorithme de routage simple-copie et multi-sauts optimum lorsque les inter-contacts sont exponentiellement distribués. Nous montrons des résultats de simulations sur les données citées précédemment. Dans ce résumé, nous ne montrons que quelques résultats concernant les données de Dartmouth.

B.1.1 Nature des distributions des temps d'inter-contacts

Nous avons étudié des données réelles provenant du réseau d'accès Wi-Fi de l'université de Dartmouth [48]. Ces données représentent les sessions des utilisateurs dans le réseau sans-fil comportant les instants d'attachement et de détachement aux différents points d'accès. Nous avons utilisé les données pré-traitées par Song et al. dans le cadre de leur étude [80] sur la prédiction de la mobilité. Nous avons testé si les processus des temps d'inter-contacts peuvent être modélisés par des processus exponentiels, en loi de puissance ou en log-normale.

Nous avons utilisé le test statistique de Cramer-Smirnov-Von-Mises [82]. Pour chaque paire de noeud (i,j), on compare la distribution cumulée I_N^{ij} des N inter-contacts observés et la supposée fonction correspondante. Les tests ont été réalisés sur les paires montrant un niveau suffisant de connectivité se caractérisant par au moins 4 contacts. le tableau B.1 présente les résultats. Nous avons testé $20{,}211$ paires et identifié parmi celles-ci 42.8% en Exponentielle, 34.2% en loi de puissance et 85.8% en log-normale. Notons qu'une distribution peut “passer” le test pour plusieurs distributions hypothétiques. Bien que l'utilisation d'un test statistique soit discutable, ce résultat nous donne une première indication de la nature de ces distributions. Celles-ci étant en majorité distribuées en log-normale.

	Dartmouth
Nombre de paires testées	20,211
Exponentielle	42.8 %
Pareto (loi de puissance)	34.2 %
Log-normale	85.8 %
Aucune	12.9 %

Table B.1: Résultats des tests statistiques.

B.1.2 Algorithmes de routage DTN

B.1.2.1 Le modèle

Considérons un réseau composé de *n* noeuds. Supposons que les instants successifs où les noeuds i et j sont en contacts tels que $t_{ij}^1 < t_{ij}^2 < t_{ij}^3 < ...$. Le temps d'inter-contact entre le k^{eme} et le $(k+1)^{eme}$ instant de contact est alors:

$$\Delta t_{ij}^k = t_{ij}^{k+1} - t_{ij}^k \tag{B.1}$$

On suppose que les Δt_{ij}^k sont des échantillons indépendants et identiquement distribués provenant d'une variable aléatoire suivant une distribution exponentielle de paramètre λ_{ij}, notée $\tau_{ij} = \mathrm{exponential}(\lambda_{ij})$. Le temps moyen d'inter-contact entre i et j est alors donné par $E[\tau_{ij}] = 1/\lambda_{ij}$.

Dans le réseau, les n noeuds ont des comportements indépendants de sorte que les $n(n-1)/2$ paires de noeuds ont des temps inter-contacts τ_{ij} suivant un processus exponentiel avec des paramètres différents. La famille des processus τ_{ij} est symétrique, on a alors $\forall i, \tau_{ii} = 0$.

Ce modèle a pour but de fournir un cadre pour l'étude de différents réseaux DTN. En particulier, ceux dont les interactions entre les noeuds sont guidées par les relations sociales. La modélisation des comportement en utilisant la distribution des temps d'inter-contacts permet de s'abstraire de la mobilité géographique des noeuds. Plusieurs hypothèses supportent le modèle: (1) les processus d'interactions entre les noeuds changent suffisamment lentement pour qu'ils soient considérés en régime stationnaire, (2) les noeuds ont des capacités de stockage illimités et les transmissions entre les noeuds sont instantanées, (3) les temps de contacts sont nuls.

B.1.2.2 Stratégies

Ici nous étudions des stratégies de routage sur le modèle que nous avons proposé ci-dessus.

Wait Dans la stratégie de routage Wait, le noeud source s attend de rencontrer la destination d pour lui transférer le message. La nature exponentielle des processus d'inter-contacts conduit pour Wait au temps moyen de livraison suivant:

$$E[D_{sd}^w] = 1/\lambda_{sd} \tag{B.2}$$

MED La stratégie Minimum Expected Delay (MED) a été introduite par Jain et al. [23]. Elle effectue un routage à la source en définissant le chemin que le message doit suivre en partant de la source s pour aller vers la destination d. La liste des relais fournie le délai minimum de bout-en-bout espéré. Le temps de livraison espéré est alors défini par:

$$E[D^{med}_{s,r_1,r_2,\ldots,r_{n-1},d}] = 1/\lambda_{sr_1} + 1/\lambda_{r_1r_2} + \ldots 1/\lambda_{r_{n-1}d} \tag{B.3}$$

Trouver le chemin optimal consiste à calculer le plus court chemin entre s et d dans un graphe valué sur chacun des liens (i,j) par $1/\lambda_{ij}$. L'algorithme de Dijkstra peut être utilisé.

Spray and Wait La stratégie Spray and Wait a été introduite par Grossglauser et Tse [28]. Elle consiste en deux étapes. D'abord, le noeud source utilise le premier noeud rencontré comme relais vers la destination. Ensuite, le noeud relais choisi attend de rencontrer la destination pour lui transmettre le message. Ici nous étudions le cas où un seul relais est utilisé, on appelle ce schéma de routage 1-SW.

Après simplification, on établit que dans un réseau composé de n noeuds, 1-SW délivre les messages entre une source s et une destination d en un temps moyen donné par:

$$E[D^{1-sw}_{sd}] = \frac{(1 + \sum_{r \neq s, r \neq d} \frac{\lambda_{sr}}{\lambda_{rd}})}{\sum_{r \neq s} \lambda_{sr}} \tag{B.4}$$

Optimum Spray and Wait Nous définissons une variation de 1-SW appelée 1-SW*. Au lieu de diffuser les messages vers le premier noeud rencontré, la source choisie un relais dans un sous-ensemble de noeuds R. Cette variante intermédiaire est nommée 1-SW^R. On définit alors 1-SW* comme la stratégie 1-SW^R utilisant le sous-ensemble R minimisant $E[D^{1-sw^R}_{sd}]$. Le temps espéré de livraison des messages est alors:

$$E[D^{1-sw^R}_{sd}] = \frac{(1 + \sum_{r \in R} \frac{\lambda_{sr}}{\lambda_{rd}})}{\sum_{r \in R} sr} \tag{B.5}$$

La version multi-sauts, appelée $SW*$, que nous avons introduite dans cette thèse est une version récursive de $1 - SW*$. Elle donne un délai optimal lorsque les temps d'inter-contacts pour chaque paire suivent une distribution exponentielle. Par manque de place, nous ne détaillons cette algorithme dans ce résumé.

B.1.3 Comparaison des routages

Cette section étudie les performances des différents algorithmes en présence d'hétérogénéité dans les distributions des temps d'inter-contacts.

Nous avons effectué des simulations sur les données de Dartmouth pour étudier le comportement des divers algorithmes de routage: Wait et 1-SW étant des stratégies dites naïves, et, 1-SW* et MED prenant en compte l'hétérogénéité des temps d'inter-contacts. Nous montrons également les résultats pour SW^* et pour le routage épidémique. Nous avons généré du trafic entre 100 différentes paires de noeuds et rejoué les interactions des 835 noeuds présents dans les données.

	delivery ratio	**M. delay** (days)	**Th. delay** (days)	**Overhead** (trans.)
Wait	8.6 ± 1.0	7.2 ± 4.4	11.9 ± 3.1	25.8 ± 3.1
1-SW	57.4 ± 2.0	14.0 ± 1.6	-	427.8 ± 15.3
1-SW*	61.4 ± 1.1	10.0 ± 0.9	8.4 ± 0.6	416.8 ± 12.8
MED	34.2 ± 1.2	15.2 ± 1.8	1.0 ± 0.1	724.8 ± 20.4
SW*	82.4 ± 1.4	4.3 ± 0.3	1.4 ± 0.1	1993.6 ± 793.4
Epidemic	99.0 ± 0.8	0.9 ± 0.0	-	123851 ± 3687.8

Table B.2: Résultats de simulation avec les données de Dartmouth.

Le tableau B.2 montre les résultats de simulation. Il présente quatre indicateurs de performances: (1) le taux de livraison, (2) le délai médian, (3) le délai théorique dans le cas ou les inter-contacts suivaient tous un processus exponentiel et (4) le nombre totale de transmissions ayant été nécessaires. Nous pouvons voir que les stratégies prenant en compte l'hétérogénéité montrent des performances bien meilleures à celles des stratégies naïves.

B.1.4 Conclusion

Dans cette étude, nous avons mis en évidence l'hétérogénéité des distributions des temps inter-contacts sur des données empiriques. Nous avons proposé un modèle simple permettant d'intégrer celle-ci puis nous avons étudié analytiquement pour quelques stratégies de routage leur performances en terme de délais de livraison. Enfin, nous avons montré au travers de simulations que les stratégies prenant en compte l'hétérogénéité sont plus efficaces que les autres.

B.2 Routage DTN basé sur les habitudes de mobilité des noeuds

Comme nous venons de le voir, dans les DTNs le routage n'est pas trivial mais le problème peut se simplifier lorsque les noeuds présentent quelques régularités dans leurs contacts ou dans les localisations qu'ils visitent. Nous proposons ici un formalisme générique pour le routage dans les DTNs utilisant un espace Euclidien construit à partir de caractéristiques liées à la mobilité des noeuds.

B.2.1 Le concept de MobySpace

Deux personnes ayant des habitudes de mobilité similaires ont une grande chance de se rencontrer et donc de pouvoir communiquer de manière ad hoc. Basé sur ce principe, nous avons proposé [9] d'utiliser le formalisme d'un espace virtuel Euclidien, appelé *MobySpace*, comme outil pour aider les noeuds à prendre les décisions de routage. L'espace virtuel est construit en fonction des habitudes de mobilité considérées et la position des noeuds dans cet espace correspond à leurs habitudes. Cette position est appelée *MobyPoint*. Le routage s'effectue alors dans l'espace virtuel en se rapprochant de noeud en noeud, en fonction des rencontres, de la position du noeud destination. De manière intuitive, on rapproche un message de sa destination en passant successivement par des noeuds ayant des habitudes de mobilité de plus en plus similaires à celle-ci. Remarquons que cette méthode n'est pas un routage géographique.

Les habitudes de mobilité peuvent être définies de plusieurs façons. Elles représentent de manière synthétique des informations sur la mobilité des noeuds ou les interactions qu'ils ont avec leur environnement. Par exemple, les habitudes de mobilité peuvent être la fréquence des contacts que les noeuds ont eu avec les autres ou la fréquence des divers endroits qu'ils visitent. Ces informations de fréquences ont pu être observées directement par les noeuds et doivent être alors partagées avec les autres, ou elles ont pu être récupérées auprès d'une entité externe réalisant des observations sur la mobilité des noeuds.

Formellement, considérons U l'ensemble des noeuds et L l'ensemble des dimensions. Le MobyPoint d'un noeud $k \in U$ est un point dans un espace à n dimensions, où $n = |L|$. Considérons $m_k = (c_{1_k}, ..., c_{n_k})$ le MobyPoint du noeud k. La distance entre deux MobyPoints est alors notée $d(m_i,m_j)$.

A chaque instant t, le noeud k possède un ensemble de voisins auquel il est directement connecté, que nous nommons $W_k(t) \subseteq U$. $W_k^+(t) = W_k(t) \cup \{k\}$ est le voisinage augmenté contenant k. Le routage MobySpace consiste soit à choisir l'un des voisins pour lui transmettre le message, soit à garder le message. La fonction de routage, appelée, choisit le voisin ayant les habitudes de mobilité les plus proches de la destination b. La décision pour le noeud k devant transmettre un message à destination de b est prise en appliquant la fonction f:

$$f(W_k^+(t),b) =$$

$$\begin{cases} b \text{ if } b \subset W_k(t) \text{, else} \\ i \in W_k^+(t) : d(m_i,m_b) = \min_{j \in W_k^+(t)} d(m_j,m_b) \end{cases} \tag{B.6}$$

Le choix de la fonction d mesurant la distance entre deux MobyPoints est important. Nous avons principalement utilisé la distance euclidienne.

B.2.2 MobySpace basé sur les fréquences de visite

Dans nos travaux, nous avons évalué un MobySpace basé sur la fréquence de visite des noeuds par rapport aux diverses localisations de l'environnement. Sur une certaine période, chaque noeud passe une proportion de son temps (pouvant être nulle) à chacune des n localisations de l'espace. L'ensemble de ces proportions constitue l'habitude de mobilité d'un noeud et décrit son MobyPoint dans un espace de dimension n. Si l'on considère les fréquences de visite comme une bonne estimation des probabilités futures, la coordonnée d'un noeud sur l'axe x est sa probabilité de visiter la localisation k. Tous les MobyPoints dans un MobySpace donné sont alors regroupés dans un hyper-plan, nous avons alors:

$$\text{for any point } m_i = (c_{1_i}, ..., c_{n_i}), \sum_{k=1}^{n} c_{k_i} = 1 \tag{B.7}$$

Des études récentes sur la mobilité d'étudiants sur un campus [52, 48] ou d'employés en entreprise [88], équipés de PDAs ou d'ordinateurs portables ont montrées des résultats similaires: de nombreux comportements sont distribués en loi de puissance. En particulier, la fréquence de visite des différents endroits: les personnes visitent très fréquemment peu de localisations alors qu'ils fréquentent beaucoup de localisations assez rarement. Henderson et al. ont observés [48] que 50% des personnes étudiées ont passé 62% de leur temps à un seul endroit et que cette proportion du temps décroît en loi de puissance pour les autres localisations.

B.2.3 Évaluation

Nous avons évalué le routage basé sur MobySpace avec les données de mobilité collectées dans le réseau d'accès Wi-fi de l'Université de Dartmouth. Ce réseau comporte 550 points d'accès placés aux différents endroits de l'université (salles de cours, bâtiments administratifs, bibliothèque, terrains de sport et logements étudiants). Environs 13000 cartes sans-fil se sont connectées entre 2001 et 2004. La plupart des utilisateurs sont des étudiants équipés de PDA ou d'ordinateurs portables (obligatoires pour suivre la scolarité). La figure B.1 montre l'évolution du nombre d'utilisateurs par jour présents dans le réseau.

Dans notre étude, chaque point d'accès représente une localisation. Nous avons fait l'hypothèse que deux noeuds peuvent communiquer en mode ad hoc avec une interface radio courte portée, du type Bluetooth par exemple, si ils sont attachés au même moment au même point d'accès. Cette hypothèse n'est pas forcément réaliste dans la mesure où l'on risque de créer des op-

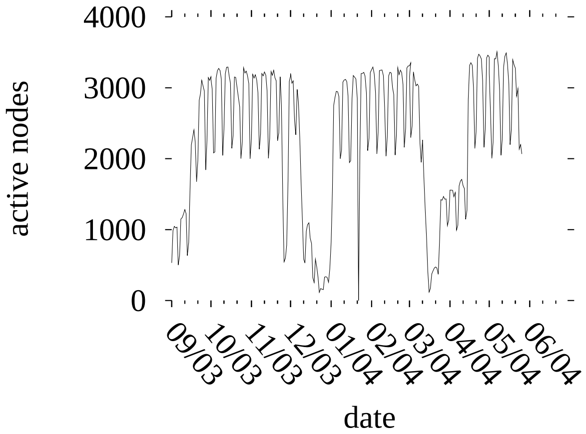


Figure B.1: Nombre d'utilisateurs par jour (du 1er septembre 2003 au 1er juin 2004).

portunités de communication artificielles. Cependant, il s'agit de l'approximation la plus simple que nous pouvons mettre en oeuvre. Jusqu'à présent, très peu de traces à large échelle sont disponibles publiquement pour des scénarios DTN. Nous espérons vivement que d'autres traces soient collectées rapidement et que nous pourrons alors tester MobySpace celles-ci.

Nous avons rejoué les traces de mobilité capturées entre le 26 janvier 2004 et le 11 mars 2004. Cette période correspondant à la période d'activité scolaire entre noël et les vacances de printemps. Ces données brutes contiennent 5545 noeuds actifs ayant visités 536 localisations. Nous montrons les résultats de simulations réalisées en rejouant uniquement la mobilité des utilisateurs étant présent tous les jours dans les données.

Nous avons comparé le routage MobySpace avec les algorithmes suivants:

- *Routage épidémique (*Epidemic*)*: Ce routage a été introduit par Vahdat and Becker [25]. Chaque fois que deux noeuds se rencontrent, ils s'échangent les messages que, respectivement, ils n'ont pas encore. Ce schéma de routage produit le délai minimum, il est alors utilisé comme borne inférieure. Cependant il consomme énormément de ressources réseaux et mémoires, et n'est donc pas, le plus souvent, réaliste.

- *Routage opportuniste (*Opportunistic*)*: Un noeud attend de rencontrer la destination pour lui transférer le message. Il s'agit du routage le plus basique et représente le cas extrême opposé au routage épidémique.

Du fait de la difficulté à simuler la mobilité de la totalité des noeuds présents dans les données, nous avons échantillonné des noeuds et réalisé plusieurs séries de simulations. Dans chaque série de simulations, la mobilité de 300 noeuds est rejouée et 100 d'entre eux sont considérés actifs. Chaque noeud actif, établit vers 5 autres noeuds une connexion consistant en l'envoi d'un message. Les résultats des différentes séries de simulations sont alors moyennés et un intervalle de confiance à 90% est fourni en utilisant la loi de Student.

	delivery ratio (%)	**delay** (days)
Epidemic	97,8 $\pm 1{,}0$	3,1 $\pm 0{,}4$
Opportunistic	10,4 $\pm 1{,}4$	19,6 $\pm 1{,}9$
MobySpace	46,6 $\pm 1{,}1$	20,2 $\pm 2{,}0$

Table B.3: Résultats de simulations.

Le tableau B.3 montre les résultats de simulations. Epidemic livre $97{,}8\%$ des messages en un délai de $3{,}1$ jours. A l'opposé, Opportunistic ne délivre que $10{,}4\%$ des messages en $19{,}6$ jours. MobySpace se situe au milieu en terme de livraison avec un taux de $46{,}6\%$ et achève un délai moyen de $20{,}2$ jours. Epidemic atteint ce fort taux de livraison en prenant avantage de toutes les opportunités de contact.

Les performances d'un routage simple-copie basé sur MobySpace sont bonnes en terme de taux de livraison mais peuvent être améliorées en utilisant plusieurs copies.

B.2.4 Stratégies multi-copies

Nous montrons ici les résultats de l'étude que nous avons réalisé autour de l'utilisation de MobySpace pour des schémas de routage multi-copies, c'est à dire des schémas dans lesquels plusieurs copies des messages circulent dans le réseau en même temps. Les routages MobySpace que nous avons étudiés s'inspirent des schémas de bases suivants:

- *Spray and Wait*: A moins de rencontrer en premier la destination d'un message, la source transmet une copie de celui-ci aux N premiers noeuds rencontrés. Ces noeuds relais transmettent alors le message à la destination si ils la rencontrent.

- *TTL based*: La source utilise le routage Epidemic mais utilise un temps de vie des messages (TTL) égal à T pour uniquement atteindre les relais situés au plus à T sauts.

- *Probabilistic flooding*: La source inonde le réseau avec un schéma du type Epidemic. Cependant, chaque relais transmet seulement une copie aux N premiers noeuds rencontrés avec une probabilité P. Lorsqu'un noeud ne doit rien retransmettre, il agit comme un relais passif comme dans Spray and Wait.

La figure B.2 montre le compromis entre la proportion des messages non-délivrés et le surcoût pour chaque classe de protocoles que nous avons évalués. Nous ne pouvons entrer

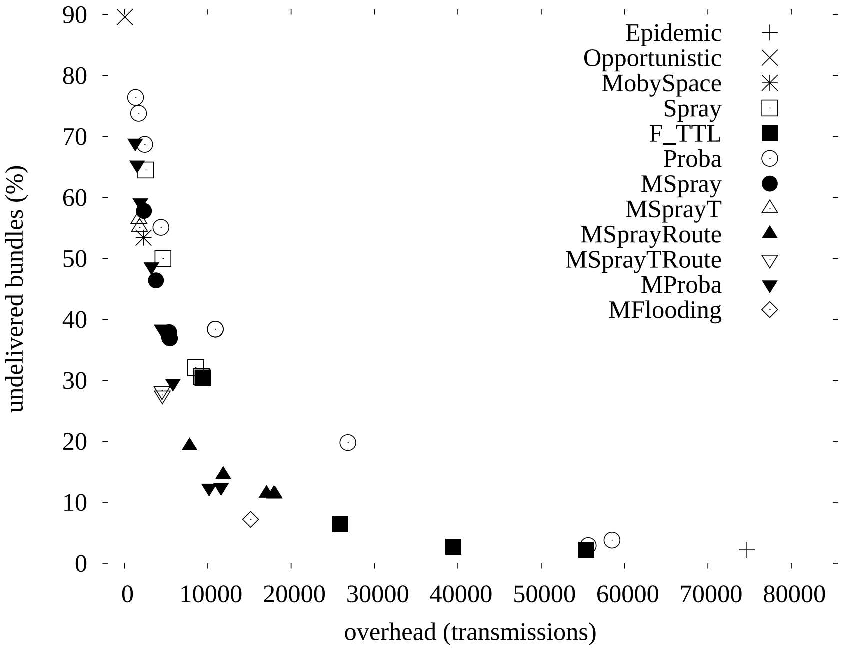


Figure B.2: Compromis entre la livraison et le surcoût.

dans le détail de toutes les stratégies testées et essayons juste ici de donner les résultats importants. Nous pouvons voir que cette courbe a une forme concave, allant de la partie en haut à gauche avec Opportunistic à la partie en bas à droite avec Epidemic. Puisque notre but est de minimiser à la fois le surcôut et le nombre de messages non délivrés, cette courbe montre que les solutions basées sur MobySpaces, comme MFlooding qui est dans le creux de la courbe, tendent à atteindre nos objectifs. MFlooding est une modification d'Epidemic, les messages sont diffusés épidémiquement mais uniquement aux noeuds nous rapprochant de la destination dans le MobySpace.

Ces résultats montrent que MobySpace peut facilement être utilisé pour des schémas de routage multi-copies et que les solutions reposant sur celui-ci ont de bonnes performances en terme de surcoût et de taux de livraison.

B.2.5 Conclusion

La contribution principale de ce travail est l'introduction d'un formalisme générique pour le routage dans les DTNs utilisant un espace virtuel euclidien, appelé MobySpace, construit pour manipuler les habitudes de mobilité des noeuds. Nous avons montré en rejouant en simulation des traces de mobilité réelles que MobySpace peut être utilisé pour définir des schémas de routage performants. Nous avons étudié des schémas simple-copie et multi-copies.

B.3 Distribution de contenu en environnement urbain

Ce travail représente notre dernière contribution. Il concerne l'étude de la faisabilité d'un système de distribution de contenu (journaux quotidiens, publicité, ...) dans un environnement urbain à l'aide de bornes sans-fil. Nous avons réalisé une expérience avec les capteurs Intel iMotes dans la ville de Cambridge, GB, afin d'étudier comment une population cible composée d'étudiants pouvait recevoir du contenu des bornes fixes.

B.3.1 Expérience

Pour l'étude de différentes stratégies de distribution de contenu, nous avons réalisé une expérience afin d'enregistrer les contacts entre une population cible d'utilisateurs mobiles et différentes localisations fixes. Cette expérience a été réalisée en utilisant les capteurs Intel iMotes composés d'une mémoire, d'un processeur et d'une interface Bluetooth. Ceux-ci enregistrent régulièrement les autres périphériques Bluetooth qu'ils rencontrent, c'est à dire les autres iMotes et les appareils (e.g., téléphones) ayant le Bluetooth activé. Nous avons donc distribué des iMotes à des étudiants de l'université de Cambridge et placé des capteurs fixes à des endroits fréquentés de Cambridge (supermarchés, pubs, centre commerciaux, ...). La figure B.3 montre la position des capteurs fixes dans la ville.

Figure B.3: Position des iMotes fixes.

Les capteurs iMotes sondent périodiquement le voisinage. Nous avons estimé cette période pour les différents types de capteurs à 2 et 6 minutes pour les fixes et 10 minutes pour les mobiles afin qu'ils aient une durée de vie de 2 semaines. Les capteurs mobiles ont été conditionnés dans de petites boites et ont été munis d'une pile d'appareil photo, de sorte qu'ils puissent rester sans gêne dans les poches des étudiants en permanence. Les iMotes fixes, ayant un conditionnement plus gros, ont été équipées de batteries ayant une capacité beaucoup plus importante.

L'expérience a officiellement démarrée le vendredi 28 octobre 2005 à 9:55:32 (GMT) et s'est arrêtée le mercredi 21 décembre 2005 à 13:00 (GMT).

B.3.2 Traces collectées

La figure B.4 montre pour les différents types de capteurs déployés (les mobiles préfixés par M, les fixes préfixés par F), leurs propriétés et quelques statistiques concernant les données qu'ils ont collectés. Nous avons récupéré 36 iMotes mobiles et 18 iMotes fixes. Les temps de vie ont été en moyenne supérieurs à 10 jours, sauf pour 2 iMotes de type FLR-2, placées à des endroits très populaires, qui ont saturé leur mémoire très rapidement. Ce tableau montre également le nombre total de contacts, le nombre de contacts dits *internes* (ayant eut lieu entre des iMotes) et le nombre de contacts dits *externes* (entre une iMote et un périphérique Bluetooth inconnu). Nous pouvons voir qu'il y a eut beaucoup d'interactions, 19014 au total, entre les étudiants. Les iMotes, tous types confondus, ont rencontrés une grande quantité de périphériques Bluetooth extérieurs. Enfin, nous n'avons pas observé un grand nombre d'interactions entre les iMotes fixes et mobiles.

	MSR-10	**FSR-10**	**FSR-6**	**FLR-2**
Nb motes	36	12	2	4
Lifetime (days)	10.7 ±0.8	11.0 ±0.6	14.5 ±0.5	15.7 ±8.3
Contacts	19014	8270	1082	11119
Int. co.	8545	38	91	102
Ext. co.	10469	8232	991	11017

Table B.4: Statistiques globales.

La figure B.4 montre l'évolution du nombre de contacts par jour entre les iMotes mobiles. Nous pouvons voir que les contacts suivent un rythme hebdomadaire, les jours 2, 3 et 9, 10 étant des samedis et des dimanches. Cette observation est naturelle puisque les étudiants étaient dans le même programme à l'université.

Un des buts de l'expérience étant de collecter des contacts entre les iMotes fixes et mobiles, la figure B.5 montre le nombre de contacts avec les iMotes mobiles pour chaque iMote fixe. Nous pouvons voir que les interactions n'ont pas été nombreuses. Ceci est sûrement dû au fait que les étudiants n'étaient pas au courants de l'emplacement des points d'accés. Les deux iMotes fixes ayant eu le plus grand nombre de contacts correspondent à la réception du laboratoire d'informatique de l'université de Cambridge et au supermarché le plus populaire de la ville.

Concernant les contacts ayant été observés avec les périphériques Bluetooth extérieurs,

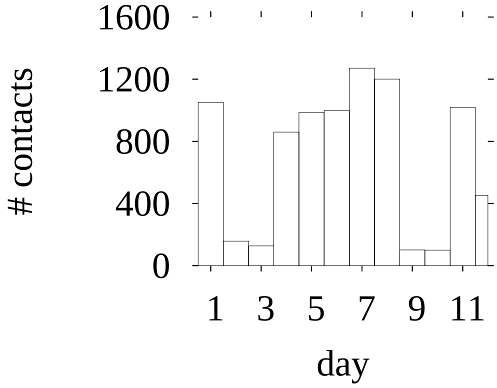


Figure B.4: Nombre de contacts par jour entre les iMotes mobiles.

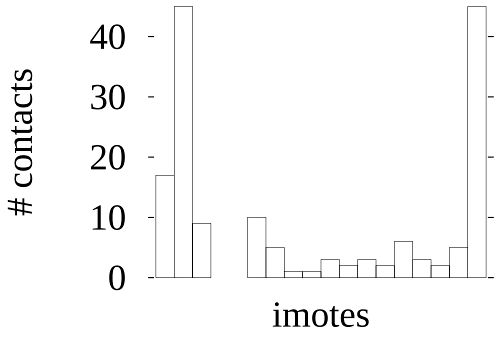


Figure B.5: Nombre contacts avec des iMotes mobiles pour les iMotes fixes.

nous avons utilisé la base de données OUI (Organizationally Unique Identifiers) [1] de l'IEEE afin d'étudier les fabriquants. Le camembert B.6 montre leur représentation pour les 97% des préfixes que nous avons résolus. Nous pouvons voir que les principaux acteurs correspondent à des fabriquants de matériel tel que des téléphones ou ordinateurs portables. Ceci montre que nous pouvons considérer les périphériques extérieurs comme des individus mobiles.

B.3.3 Scénario étudié

A l'aide des données recueillies, nous avons pu étudier un scénario dans lequel un journal est distribué à partir de 7 heures tous les matins via les bornes fixes Bluetooth (simulées par les iMotes fixes). Notre population cible, présumée intéressée par le journal, est représentée par les étudiants transportant dans leur poche les iMotes mobiles. Le but de ce scénario est d'étudier les différentes stratégies de distribution. Nous avons rejoué 5 jours de données, du lundi au

---

[1] http://standards.ieee.org/regauth/oui/

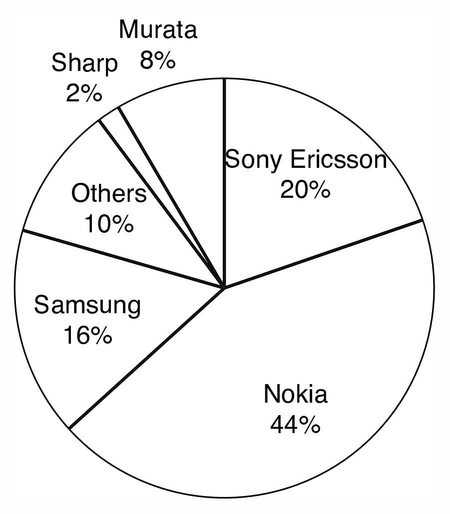


Figure B.6: Manufacturers

vendredi et avons fait l'hypothèse que lorsque les noeuds mobiles ont récupérés une copie du journal, il la conserve jusqu'au lendemain matin 7 heures. Nous avons simulé les stratégies de distribution suivantes:

- *Egoïste*: Les noeuds passant à coté des bornes récupèrent une copie du journal mais ne la retransmettent pas aux autres noeuds mobiles.
- *Communautaire*: Les noeuds récupèrent le contenu en passant près des bornes et ont la possibilité de le partager avec les autres noeuds de la communauté (i.e., les autres étudiants).
- *Collaboration étendue*: En plus d'utiliser la stratégie communautaire, ce mode de distribution utilise des périphériques extérieurs dans le processus de partage. Dans ce cadre, des politiques incitant les gens à partager le contenu doivent être mises en place mais n'ont pas été abordées dans nos travaux.

Afin de mesurer les performances des différentes approches, nous avons utilisé les métriques suivantes:

- *Delivery ratio*: le pourcentage d'utilisateurs (les étudiants) satisfaits, c'est à dire ayant reçu une copie du journal avant 7 heures le lendemain.
- *Average delay*: Le temps moyen de livraison du journal.

- *Efficiency*: Le nombre de messages transmis par copie délivrée du journal.

	Delivery	**Delay**
Selfish	20,5	7,47
Collectivist	90,2	5,29
All external	97,1	4,10

Table B.5: Résultats de simulation.

La figure B.5 présente les résultats de simulation. La première chose que l'on peut observer est que la stratégie Égoïste ne permet d'atteindre qu'un taux de satisfaction de $20{,}5$%. Cela semble naturel dans la mesure où il n'y a pas eu beaucoup de contacts entre les iMotes fixes et mobiles. Nous observons, une nette amélioration lorsque les étudiants partagent les copies qu'ils récupèrent dans le mode communautaire avec $90{,}2$% de satisfaction. En parallèle, nous pouvons remarquer que le mode communautaire permet de baisser le délai moyen de livraison de $7{,}47$ heures à $5{,}29$ heures. Le mode de collaboration étendue impliquant tous les périphériques dans le processus de partage permet d'atteindre $97{,}1\%$ de statisfaction.

B.3.4 Conclusion

Nous avons proposé dans ce travail une expérience avec des iMotes permettant l'étude d'un scénario de distribution de contenu inédit. Nous avons pu montrer qu'une stratégie basée sur un partage communautaire permet d'atteindre de bonnes performances. Les traces collectées sont désormais accessibles sur le site CRAWDAD pour d'autres recherches.

Appendix C

Conclusion

Le domaine des réseaux tolérants aux délais (DTN) a réellement émergé ces deux dernières années afin de fournir des mécanismes permettant d'étendre l'architecture de l'Internet actuel. Les réseaux abordés par ce domaine ont en commun le fait que leur connectivité est perturbée ou que le niveau d'hétérogénéité est tel que les protocoles usuels de l'Internet ne fonctionnent plus.

Cette thèse représente nos contributions pour les DTNs. Nous avons réalisé des études sur:

- *La caractérisation des interactions entre les noeuds*: Mieux comprendre les interactions entre les noeuds DTN permet de proposer des schémas de routage efficaces. Dans ce but, nous avons proposé une modélisation inédite intégrant l'hétérogénéité des interactions entre les paires de noeuds à partir d'observations de données réelles.

- *Routage et distribution de contenu*: Nous avons proposé divers schémas de routage. Les deux premiers chapitres ont proposés des routages basés sur la connaissance d'informations concernant l'historique de la mobilité des noeuds. Dans le dernier chapitre nous avons étudié une solution de distribution de contenu en environnement urbain.

- *Collecte de traces*: Les modèles de mobilité dans les DTNs étant encore peu réalistes, des efforts ont besoin d'être conduits pour collecter des traces en conditions réelles. Nous avons contribué à cet effort avec l'expérience que nous avons réalisée à Cambridge.

Nous pensons que nos contributions ont un fort intérêt pour la communauté DTN. Nous nous sommes positionnés parmi les précurseurs dans l'utilisation de traces d'interactions réelles. Nous avons proposé et évalué des stratégies de routage sur ces traces et nous avons activement participé à l'effort de collection.

Les perspectives de travail permettant d'étendre ces recherches sont nombreuses. En dépit du peu de traces disponibles publiquement, les propositions de routage exposés dans cette thèse peuvent être évaluées sur d'autres données comme celles du projet Reality Mining du MIT [90] ou celles du projet DieselNet [67] de UMass. Par ailleurs, d'autres études analytiques doivent être menées afin de prolonger l'effort de compréhension de la mobilité des noeuds DTN et de proposer des modèles plus réalistes intégrant d'autres facteurs. De nouvelles solutions de routage peuvent également être proposées et comparées avec celles existantes dans la littérature. Enfin, d'autres expériences peuvent être menées afin de collecter des traces utiles à la communauté.

Bibliography

[1] R. Shah, S. J. S. Roy, and W. Brunette, “Data mules: Modeling a three-tier architecture for sparse sensor networks,” Intel Research Seattle, Tech. Rep. IRS-TR-03-001, January 2003.

[2] S. Corson, “Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations,” RFC 2501. IETF, January 1999.

[3] J. Scott, P. Hui, J. Crowcroft, and C. Diot, “Haggle: A networking architecture designed around mobile users,” in *Proc. WONS*, 2006.

[4] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot, “Pocket switched networks and human mobility in conference environments,” in *Proc. WDTN*, 2005.

[5] K. Fall, “A delay-tolerant network architecture for challenged internets,” Intel Research Berkeley, Tech. Rep. IRB-TR-03-003, February 2003.

[6] “Inter-planetary internet,” http://www.ipnsig.org.

[7] I. F. Akyildiz, O. Akan, C. Chen, J. Fang, and W. Su, “Interplanetary internet: state-of-the-art and research challenges,” *Computer Networks*, vol. 43, no. 2, pp. 75–112, 2003.

[8] V. Conan, J. Leguay, and T. Friedman, “The heterogeneity of inter-contact time distributions: its tance for routing in delay tolerant networks,” Arxiv, Tech. Rep. cs.NI/0609068, October 2006.

[9] J. Leguay, T. Friedman, and V. Conan, “DTN routing in a mobility pattern space,” in *Proc. WDTN*, 2005.

[10] J. Leguay, , T. Friedman, and V. Conan, “Evaluating mobility pattern space routing for DTNs,” in *Proc. INFOCOM*, 2006.

[11] J. Leguay, A. Lindgren, J. Scott, T. Friedman, and J. Crowcroft, “Opportunistic content distribution in an urban setting,” in *Proc. (CHANTS)*, 2006.

[12] K. Fall, “A delay-tolerant network architecture for challenged internets,” in *Proc. SIGCOMM*, 2003.

[13] “Delay Tolerant Network Research Group (DTNRG),” http://www.dtnrg.org.

[14] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss, “Delay tolerant network architecture, IRTF draft, draft-irtf-dtnrg-arch-02.txt,” July 2004.

[15] K. Scott and S. Burleigh, “Bundle protocol specification, IRTF draft, draft-irtf-dtnrg-bundle-spec-06.txt,” August 2006.

[16] F. Warthman, “Delay-tolerant networks (dtns): A tutorial v1.1,” March 2003.

[17] R. Stewart, Q. Xie, K. Morneault, C. Sharp, H. Schwarzbauer, T. Taylor, I. Rytina, M. Kalla, L. Zhang, and V. Paxson, “Stream control transmission protocol,” RFC 2960. IETF, October 2000.

[18] D. Waitzman, “A standard for the transmission of ip datagrams on avian carriers,” RFC 2549. IETF, April 1990.

[19] A. Seth, P. Darragh, S. Liang, Y. Lin, and S. Keshav, “An architecture for tetherless communication,” in *Under submission*, July 2005.

[20] A. Seth, S. Bhattacharyya, , and S. Keshav, “Application support for opportunistic communication on multiple wireless networks,” in *Under submission*, 2005.

[21] J. Ott and D. Kutscher, “A disconnection-tolerant transport for drive-thru internet environment,” in *Proc. INFOCOM*, 2005.

[22] “OCMP,” http://blizzard.cs.uwaterloo.ca/tetherless/index.php/OCMP.

[23] S. Jain, K. Fall, and R. Patra, “Routing in a delay tolerant network,” in *Proc. SIGCOMM*, 2004.

[24] E. P. Jones and P. A. Ward, “Routing strategies for delay-tolerant networks,” in *Submitted to Computer Communication Review*, 2006.

[25] A. Vahdat and D. Becker, “Epidemic routing for partially connected ad hoc networks,” Duke University, Tech. Rep. CS-200006, April 2000.

[26] H. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D. Rubenstein, “Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with Zebranet,” in *Proc. ASPLOS-X*, 2002.

[27] M. Grossglauser and D. N. C. Tse, “Mobility increases the capacity of ad-hoc wireless networks,” in *INFOCOM*, 2001.

[28] M. Grossglauser and D. Tse, “Mobility increases the capacity of ad-hoc wireless networks,” *Transactions on Networking*, vol. 10, no. 4, pp. 477–486, August 2002.

[29] T. Spyropoulos, K. Psounis, and C. Raghavendra, “Spray and wait: An efficient routing scheme for intermittently connected mobile networks,” in *Proc. WDTN*, 2005.

[30] K. Harras, K. Almeroth, and E. Belding-Royer, “Delay tolerant mobile networks (DTMNs): Controlled flooding schemes in sparse mobile networks,” in *Proc. Netwoking*, 2005.

[31] Y. Wang, S. Jain, M. Martonosi, and K. Fall, “Erasure coding based routing for opportunistic networks,” in *Proc. WDTN*, 2005.

[32] J. Widmer and J. L. Boudec, “Network coding for efficient communication in extreme networks,” in *Proc. WDTN*, 2005.

[33] A. Lindgren, A. Doria, and O. Schelen, “Probabilistic routing in intermittently connected networks,” in *Proc. SAPIR*, 2004.

[34] A. Lindgren and A. Doria, “Probabilistic routing protocol for intermittently connected networks, IRTF draft, draft-lindgren-dtnrg-prophet-02.txt,” March 2006.

[35] “The sami network connectivity (SNC) project,” http://www.snc.sapmi.net.

[36] B. Burns, O. Brock, and B. N. Levine, “MV routing and capacity building in disruption tolerant networks,” in *Proc. INFOCOM*, 2005.

[37] J. A. Davis, A. H. Fagg, and B. N. Levine, “Wearable computers as packet transport mechanisms in highly-partitioned ad-hoc networks,” in *Proc. ISWC*, 2001.

[38] M. Musolesi, S. Hailes, and C. Mascolo, “Adaptive routing for intermittently connected mobile ad hoc networks,” in *Proc. WOWMOM*, 2005.

[39] J. LeBrun, C. Chuah, and D. Ghosal, “Knowledge based opportunistic forwarding in vehicular wireless ad hoc networks,” in *Proc. VTC Spring*, 2005.

[40] E. P. C. Jones, L. Li, and P. A. S. Ward, “Practical routing in delay-tolerant networks,” in *Proc. WDTN*, 2005.

[41] “Sensor networking with delay tolerance (SeNDT),” http://down.dsg.cs.tcd.ie/sendt/.

[42] W. Zhao, M. Ammar, and E. Zegura, “A message ferrying approach for data delivery in sparse mobile ad hoc networks,” in *Proc. ACM Mobihoc*, 2004.

[43] Zhao, M. Ammar, and E. Zegura, “Controlling the mobility of multiple data transport ferries in a DTN,” in *Proc. INFOCOM*, 2005.

[44] S. Jain, M. Demmer, R. Patra, and K. Fall, “Using redundancy to cope with failures in a delay tolerant network,” in *Proc. SIGCOMM*, 2005.

[45] “CRAWDAD a Community Ressource for Archiving Wireless Data At Dartmouth,” http://crawdad.cs.dartmouth.edu. [Online]. Available: http://crawdad.cs.dartmouth.edu

[46] N. Sarafijanovic-Djukic, M. Piorkowski, and M. Grossglauser, “Island hopping: Efficient mobility-assisted forwarding in partitioned networks,” in *Proc. IEEE SECON*, 2006.

[47] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, “Pocket switched networks: Real-world mobility and its consequences for opportunistic forwarding,” Intel Research Cambridge, Tech. Rep. UCAM-CL-TR-617, February 2005.

[48] T. Henderson, D. Kotz, and I. Abyzov, “The changing usage of a mature campus-wide wireless network,” in *Proc. MobiCom*, 2004.

[49] M. Kim and D. Kotz, “Classifying the mobility of users and the popularity of access points,” in *Proc. LoCA*, 2005.

[50] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, “Impact of human mobility on the design of opportunistic forwarding algorithms,” in *Proc. INFOCOM*, 2006.

[51] C. Tuduce and T. Gross, “A mobility model based on wlan traces and its validation,” in *Proc. INFOCOM*, 2005.

[52] M. McNett and G. M. Voelker, “Access and mobility of wireless PDA users,” UC San Diego, Tech. Rep., 2004.

[53] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, “MaxProp: Routing for vehicle-based disruption tolerant networking,” in *Proc. INFOCOM*, 2006.

[54] N. Eagle and A. Pentland, “Reality mining: Sensing complex social systems,” *Personal and Ubiquitous Computing*, 2005.

[55] V. Srinivasan, M. Motani, and W. T. Ooi, “Analysis and implications of student contact patterns derived from campus schedules,” in *Proc. MobiCom*, 2006.

[56] A. Einstein, *Investigations on the Theory of the Brownian Movement*. Dover Publications, 1906.

[57] D. B. Johnson and D. A. Maltz, “Dynamic source routing in ad hoc wireless networks,” in *Mobile Computing*, Imielinski and Korth, Eds. Kluwer Academic Publishers, 1996, vol. 353. [Online]. Available: citeseer.ist.psu.edu/johnson96dynamic.html

[58] V. Borrel, M. Amorim, and S. Fdida, “On natural mobility models,” in *Proc. WAC*, 2005.

[59] M. Musolesi and C. Mascolo, “A Community based Mobility Model for Ad Hoc Network Research,” in *Proc. REALMAN*, 2006.

[60] J. Kim and S. Bohacek, “A survey-based mobility model of people for simulation of urban mesh networks,” in *Proc. MeshNets*, 2005.

[61] F. Legendre, V. Borrel, M. Amorim, and S. Fdida, “Reconsider microscopic mobility modeling for self-organizing networks,” *IEEE Communications Magazine*, December 2006.

[62] “BUZZeeBee,” http://www.inventop.com/.

[63] “MobiLuck,” http://www.mobiluck.co.uk.

[64] “ProxiDating,” http://www.proxidating.com/.

[65] “Socialight,” http://socialight.com/.

[66] M. Lukac, L. Girod, and D. Estrin, “Disruption tolerant shell,” in *Proc. CHANTS*, 2006.

[67] “UMassDieselNet: A Bus-based Disruption Tolerant Network,” http://prisms.cs.umass.edu/diesel/.

[68] J. LeBrun and C. Chuah, “Bluetooth content distribution stations on public transit,” in *Proc. MobiShare*, 2006.

[69] “The cartel,” http://cartel.csail.mit.edu/.

[70] G. Karlsson, V. Lenders, , and M. May, “Delay-tolerant broadcasting,” in *Proc. ACM SIGCOMM Workshop on Challenged Networks (CHANTS)*, 2006.

[71] “Rollernet: Analysis and use of mobility in rollerblade tours,” http://www-rp.lip6.fr/rollernet/.

[72] F. Benbadis, J. Leguay, V. Borrel, M. Dias de Amorim, and T. Friedman, “Millipede: A Rollerblade Positioning System,” in *Proceedings of ACM WINTECH 2006*, 2006.

[73] “Diverse Outdoor Mobile Environment (DOME),” http://prisms.cs.umass.edu/dome/.

[74] A. Seth, D. Kroeker, M. Zaharia, S. Guo, and S. Keshav, “Low-cost communication for rural internet kiosks using mechanical backhaul,” in *Proc. MOBICOM*, 2006.

[75] “Tier: Technology and infrastucture for emerging regions,” http://tier.cs.berkeley.edu/.

[76] M. Demmer, E. Brewer, K. Fall, S. Jain, M. Ho, and R. Patra, “Implementing delay tolerant networking,” Intel Research Berkeley, Tech. Rep. IRB-TR-04-020, December 2004.

[77] N. Sarafijanovic-Djukic and M. Grossglauser, “Last encounter routing under random waypoint mobility,” in *Networking*, 2004.

[78] T. Spyropoulos, K. Psounis, and C. Raghavendra, “Single-copy routing in intermittently connected mobile networks,” in *Proc. IEEE SECON*, 2004.

[79] Spyropoulos, K. Psounis, and C. Raghavendra, “Multi-copy routing in intermittently connected mobile networks,” USC,” Technical Report, 2004.

[80] L. Song, D. Kotz, R. Jain, and X. He, “Evaluating location predictors with extensive Wi-Fi mobility data,” in *Proc. INFOCOM*, 2004.

[81] J. Yoon, B. Noble, M. Liu, and M. Kim, “Building realistic mobility models from coarse-grained traces,” in *Proc. MobiSys*, 2006.

[82] W. Eadie, *Statistical Methods in Experimental Physics*. Elsevier Science Ltd, 1971.

[83] I. Carreras, D. Miorandi, and I. Chlamtac, “A framework for opportunistic forwarding in disconnected networks,” in *Proc. of MOBIQUITOUS*, 2006.

[84] P. Bremaud, *Markov Chains, Gibbs Fields, Monte Carlo simulation, and queues*. Springer, 1999.

[85] J.-M. Francois and G. Leduc, "Predictable disruption tolerant networks and delivery guarantees," Arxiv, Tech. Rep. cs.NI/0612034, 2006.

[86] X. Zhang, G. Neglia, J. Kurose, and D. Towsley, "Performance modeling of epidemic routing," in *Proc. of IFIP Networking*, 2006.

[87] Y. Liao, K. Tan, Z. Zhang, and L. Gao, "Modeling redundance-based routing in delay tolerant networks," in *Proc. CCNC*, 2007.

[88] M. Balazinska and P. Castro, "Characterizing Mobility and Network Usage in a Corporate Wireless Local-Area Network," in *Proc. MobiSys*, 2003.

[89] C. Lonvick, "The BSD syslog protocol," RFC 3164. IETF, August 2001.

[90] N. Eagle and A. Pentland, "Social serendipity: Mobilizing social software," in *Proc. PerCom*, 2005.

[91] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, and K. Scott, "Delay-tolerant networking: an approach to interplanetary internet," *IEEE Communications Magazine*, vol. 41, no. 6, pp. 128–136, 2003.

[92] T. Small and Z. J. Haas, "Resource and performance tradeoffs in delay-tolerant wireless networks," in *Proc. WDTN*, 2005.

[93] A. Doria, M. Uden, and D. P. Pandley, "Providing connectivity to the Saami nomadic community," in *Proc. Developement by Design Conference*, 2002.

[94] J. Lawrence and T. Payne, "Exploiting familiar strangers: creating a community content distribution network of co-located individuals," in *Proc. FOAF*, 2004.

[95] E. Paulos and E. Goodman, "The familiar stranger: Anxiety, comfort, and play in public places," in *Proc. ACM SIGCHI*, 2004.

[96] J. Ghosh, M. J. Beal, H. Q. Ngo, and C. Qiao, "On profiling mobility and predicting locations of campus-wide wireless network users," State University of New York at Buffalo, Tech. Rep. CSE-2005-27, December 2005.

[97] J. Ott, D. Kutscher, and C. Dwertmann, "Integrating DTN and MANET routing," in *Proc. CHANTS*, 2006.

[98] A. Lindgren, C. Diot, and J. Scott, "Impact of communication infrastructure on forwarding in pocket switched networks," in *Proc. (CHANTS)*, 2006.

[99] C. Perkins, E. Belding-Royer, and S. Das, “RFC 3561: Ad hoc on-demand distance vector (AODV) routing,” IETF, July 2003.

[100] T. Clausen and P. Jacquet, “RFC 3626: Optimized link state routing protocol (OLSR),” IETF, October 2003.

Glossary

ADSL Asymmetric Digital Subscriber Line, 3

ADUs Application Data Units, 11

AODV Ad hoc On Demand Distance Vector, 16

AP Access Point, 21

API Application Programming Interface, 15

CRAWDAD Community Resource for Archiving Wireless Data At Dartmouth, 20

DARPA Defense Advanced Research Projects Agency, 2

DHT Distributed Hash Table, 14

DTNRG Delay Tolerant Network Research Group, 2

FTP File Transfer Protocol, 15

GMT Greenwich Mean Time, 89

GPS Global Positioning System, 19

GSM Global System for Mobile Communications, 21

GUID Globally Unique Identifier, 14

HTTP Hypertext Transfer Protocol, 15

IPN Inter-Planetary Network, 2

IRTF Internet Research Task Force, 2

MANETs Mobile Ad hoc Network, 4

MEED Minimum Estimated Expected Delay, 19

OCMP Opportunistic Connection Management Protocol, 14

OLSR Optimized Link State Routing Protocol, 16

PCMP Persistent Connectivity Management Protocol, 14

PDA Personal Data Agent, 1

POP Post Office Protocol, 6

PSNs Pocket Switched Networks, 4

SCTP Stream Control Transmission Protocol, 12

TCA Tetherless Communication Architecture, 14

TCP Transmission Control Protocol, 2

TTL Time To Live, 73

UDP User Datagram Protocol, 2

UMTS Universal Mobile Telecommunications System, 1

UWSNs Underwater Sensor Networks, 5

VANETs Vehicular Ad Hoc Networks, 5

VHF Very High Frequency, 24

Wi-Fi Wireless Fidelity, 21

WSNs Wireless Sensor Networks, 5

List of Figures

List of Tables