

Πανεπιστήμιο Ιωαννίνων  
Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής

Πτυχιακή Εργασία

Έλεγχος Συμφόρησης με Ρυθμιζόμενη  
Δικαιοσύνη σε Οπορτουνιστικά Δίκτυα

Δημήτριος-Γεώργιος Ακαστορίδης

Επιβλέπων:

Επίκουρος Καθηγητής

Ευάγγελος Παπαπέτρου

Οκτώβριος 2013

University of Ioannina  
Department of Computer Science and Engineering

Bachelor's Thesis

# Congestion Control with Adjustable Fairness in Opportunistic Networks

**Dimitrios-Georgios Akestoridis**

Advisor:  
Assistant Professor  
Evangelos Papapetrou

October 2013

# Dedication

*I dedicate this thesis to Katerina, who supported and encouraged me throughout the course of this work.*

# Acknowledgements

First of all, I would like to thank my advisor, Assistant Professor Evangelos Papapetrou, for his guidance throughout this research work and for the valuable knowledge that I acquired during our cooperation. Besides my advisor, I would also like to thank the Ph.D. candidate Nikolaos Papanikos for our fruitful discussions and his useful suggestions, even under difficult times. Finally, I am also grateful to all my teachers during my undergraduate studies who helped me shape my way of thinking.

# Abstract

Dimitrios-Georgios Akestoridis.

Bachelor of Science, Department of Computer Science and Engineering, University of Ioannina, October 2013.

Congestion Control with Adjustable Fairness in Opportunistic Networks.

Advisor: Assistant Professor Evangelos Papapetrou.

In Opportunistic Networks, where the topology is stochastic and unknown, the nodes have to seize the opportunities of forwarding data to other nodes, in order to deliver them to their destinations successfully. For this reason, the routing protocols that have been proposed for these networks use various techniques to determine the suitable data carriers. However, it has been observed that the performance of these networks depends heavily on a small subset of important nodes, which results in an unbalanced traffic load distribution. These nodes are usually under congestion, which leads to packet drops due to the storage constraints.

In this thesis we study the problem of congestion control in Opportunistic Networks and we propose a new congestion control mechanism with adjustable fairness, which provides a trade-off between efficiency and overhead. The proposed mechanism achieves high delivery ratio and low delay, without excessive use of the most important nodes, by taking into account the destination of each message and the saturation state of each node. Finally, we show that we can significantly improve the performance of the network with a slight distortion of fairness.

**Keywords:** Congestion Control, Fairness, Routing, Opportunistic Networks, Delay-Tolerant Networks, Intermittently Connected Networks.

# Εκτεταμένη Περίληψη

Δημήτριος-Γεώργιος Αλεσπορίδης του Δημητρίου και της Ελευθερίας.

Πτυχίο, Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής, Πανεπιστήμιο Ιωαννίνων, Οκτώβριος 2013.

Έλεγχος Συμμόρφωσης με Ρυθμιζόμενη Δικαιοσύνη σε Οπορτουμιστικά Δίκτυα.

Επιβλέπων: Επίκουρος Καθηγητής Ευάγγελος Παπαπέτρου.

Τα πρώτα Ασύρματα Δίκτυα Υπολογιστών υλοποιήθηκαν τη δεκαετία του 1970, ενώ τα πρώτα προϊόντα Ασύρματων Τοπικών Δικτύων έγιναν διαθέσιμα στην αγορά στις αρχές της δεκαετίας του 1990. Ωστόσο, η απουσία ενός κοινού προτύπου είχε ως αποτέλεσμα τα περισσότερα από αυτά τα προϊόντα να είναι σπάνια συμβατά μεταξύ τους. Η πρώτη έκδοση του προτύπου IEEE 802.11 προτάθηκε το 1997 ορίζοντας δύο βασικούς τρόπους λειτουργίας, τη λειτουργία υποδομής όπου οι χρήστες επικοινωνούν μέσω ασύρματων σημείων πρόσβασης και την κατά περίπτωση λειτουργία όπου οι χρήστες επικοινωνούν απευθείας μεταξύ τους. Τα Κινητά Κατά Περίπτωση Δίκτυα επεκτείνουν την κατά περίπτωση λειτουργία, παρέχοντας τη δυνατότητα επικοινωνίας πολλών αλμάτων, καθώς ο κάθε χρήστης μπορεί να λειτουργήσει ως δρομολογητής δεδομένων.

Για την επιτυχή παράδοση των δεδομένων στα Κινητά Κατά Περίπτωση Δίκτυα είναι απαραίτητη η ύπαρξη συνεχούς συνδεσιμότητας, η οποία ωστόσο μπορεί να μην υφίσταται σε ορισμένες περιπτώσεις. Στα Δίκτυα Διακοπτόμενης Συνδεσιμότητας δεν υπάρχει καμία εγγύηση ότι ένα πλήρως συνδεδεμένο μονοπάτι μεταξύ οποιουδήποτε ζεύγους χρηστών υφίσταται σε οποιαδήποτε χρονική στιγμή. Εφαρμογές αυτών των δικτύων αποτελούν τα Δίκτυα Ανεκτικά σε Καθυστέρηση, τα οποία μπορούν να ανεχθούν καθυστερήσεις μεγάλης χρονικής διάρκειας. Προκειμένου να αντιμετωπιστεί το πρόβλημα της συνδεσιμότητας, οι χρήστες πρέπει να είναι σε θέση να κρατήσουν αποθηκευμένα τα δεδομένα που λαμβάνουν για μεγάλα χρονικά διαστήματα, έως ότου βρεθεί ο κατάλληλος χρήστης για να τα προωθήσουν.

Επιπλέον, τα Δίκτυα Διακοπτόμενης Συνδεσιμότητας μπορούν να κατηγοριοποιηθούν με βάση τη γνώση της τοπολογίας του δικτύου.

Στα Οπορτουμιστικά Δίκτυα, όπου η τοπολογία είναι στοχαστική και άγνωστη, οι κόμβοι πρέπει να αδράξουν τις ευκαιρίες προώθησης δεδομένων σε άλλους κόμβους, προκειμένου να τα παραδώσουν στους προορισμούς τους επιτυχώς. Για αυτόν τον λόγο, τα πρωτόκολλα δρομολόγησης που έχουν προταθεί για αυτά τα δίκτυα χρησιμοποιούν διάφορες τεχνικές για να καθορίσουν τους κατάλληλους μεταφορείς δεδομένων. Ωστόσο, έχει παρατηρηθεί ότι η απόδοση αυτών των δικτύων εξαρτάται σε μεγάλο βαθμό από ένα μικρό υποσύνολο σημαντικών κόμβων, το οποίο έχει ως αποτέλεσμα μία μη ισορροπημένη κατανομή του κυκλοφοριακού φόρτου. Αυτοί οι κόμβοι είναι συνήθως υπό συμφόρηση, η οποία οδηγεί σε απώλειες πακέτων λόγω των αποθηρευτικών περιορισμών.

Στην παρούσα εργασία μελετάμε το πρόβλημα του ελέγχου της συμφόρησης σε Οπορτουμιστικά Δίκτυα και προτείνουμε έναν νέο μηχανισμό ελέγχου συμφόρησης με ρυθμιζόμενη δικαιοσύνη, ο οποίος παρέχει τη δυνατότητα συμβιβασμού μεταξύ αποδοτικότητας και επιβάρυνσης. Ο προτεινόμενος μηχανισμός επιτυγχάνει υψηλή αναλογία παράδοσης και χαμηλή καθυστέρηση, χωρίς υπερβολική χρήση των πιο σημαντικών κόμβων, λαμβάνοντας υπόψη τον προορισμό του κάθε μηνύματος και την κατάσταση κορεσμού του κάθε κόμβου. Τέλος, δείχνουμε ότι μπορούμε να βελτιώσουμε σημαντικά την απόδοση του δικτύου με μία μικρή διαστρέβλωση της δικαιοσύνης.

**Λέξεις Κλειδιά:** Έλεγχος Συμφόρησης, Δικαιοσύνη, Δρομολόγηση, Οπορτουμιστικά Δίκτυα, Δίκτυα Ανεκτικά σε Καθυστέρηση, Δίκτυα Διακοπτόμενης Συνδεσιμότητας.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Opportunistic Networking . . . . .	1
1.2	Scope of the Thesis . . . . .	4
1.3	Overview of the Thesis . . . . .	5
<b>2</b>	<b>Background and Related Work</b>	<b>6</b>
2.1	Routing in Opportunistic Networks . . . . .	6
2.1.1	Flooding-Based Routing Protocols . . . . .	7
2.1.2	History-Based Routing Protocols . . . . .	8
2.1.3	Social-Based Routing Protocols . . . . .	9
2.2	Coping with Resource Constraints . . . . .	10
2.2.1	Strategies for Single-Copy Routing Protocols . . . . .	11
2.2.2	Strategies for Multi-Copy Routing Protocols . . . . .	12
<b>3</b>	<b>Congestion Control in Opportunistic Networks</b>	<b>13</b>
3.1	The Need for Adjustable Fairness . . . . .	13
3.2	Congestion Control with Adjustable Fairness . . . . .	15
3.3	Exploiting Social Preferences and Heterogeneity . . . . .	18
<b>4</b>	<b>Experimental Evaluation</b>	<b>20</b>
4.1	Simulation Setup . . . . .	20
4.2	Evaluation Criteria . . . . .	22
4.3	Performance Analysis . . . . .	23
4.3.1	Simulations on the Reality Mining Data Set . . . . .	25
4.3.2	Simulations on the PMTR and Sassy Data Sets . . . . .	28

<b>5</b>	<b>Conclusions and Future Work</b>	<b>32</b>
5.1	Conclusions . . . . .	32
5.2	Future Work . . . . .	33
	<b>Bibliography</b>	<b>34</b>
	<b>Appendix A The SimBet Algorithm</b>	<b>47</b>
	<b>Appendix B The FairRoute Algorithm</b>	<b>51</b>
	<b>Appendix C The Impact of Social Preferences</b>	<b>53</b>
	<b>Short Vita</b>	<b>57</b>

# List of Figures

- 1.1 An example of data delivery in an Intermittently Connected Network . . . . . 3
- 2.1 An illustration of (a) Single-Copy and (b) Multi-Copy Routing . . . . . 6
- 3.1 Distribution of Forwards without resource constraints . . . . . 14
- 3.2 Two nodes in contact with known utility values and buffer contents . . . . . 16
- 3.3 The impact of  $\delta$  on the forwarding condition . . . . . 17
- 4.1 (a) Average Delay, (b) Cumulative Distribution Function of the Delay, (c) Delivery Ratio, and (d) Average Number of Hops on the Reality Mining data set . . . . . 26
- 4.2 (a) Total Number of Packet Drops, (b) Cumulative Fraction of Packet Drops, (c) Overhead, and (d) Cumulative Fraction of Forwards on the Reality Mining data set . . . . . 27
- 4.3 Average Delay vs. Delivery Ratio on the Reality Mining data set . . . . . 28
- 4.4 (a) Average Delay, (b) Cumulative Distribution Function of the Delay, (c) Cumulative Fraction of Packet Drops, and (d) Average Delay vs. Delivery Ratio on the PMTR data set . . . . . 29
- 4.5 (a) Average Delay, (b) Cumulative Distribution Function of the Delay, (c) Cumulative Fraction of Packet Drops, and (d) Average Delay vs. Delivery Ratio on the Sassy data set . . . . . 30
- A.1 The gray area of the given social graph corresponds to the ego network of the black node . . . . . 48
- A.2 Exchange of Messages in SimBet . . . . . 49

C.1	The impact of $\delta$ : (a) Average Delay, (b) Cumulative Distribution Function of the Delay, (c) Delivery Ratio, and (d) Average Number of Hops . . . . .	54
C.2	The impact of $\delta$ : (a) Total Number of Packet Drops, (b) Cumulative Fraction of Packet Drops, (c) Overhead, and (d) Cumulative Fraction of Forwards	55
C.3	The impact of $\delta$ on the Average Delay vs. Delivery Ratio . . . . .	56

# List of Tables

4.1	Characteristics of the data sets . . . . .	20
4.2	Simulation Settings . . . . .	21
4.3	The forwarding condition of each algorithm . . . . .	24

# Chapter 1

## Introduction

### 1.1 Opportunistic Networking

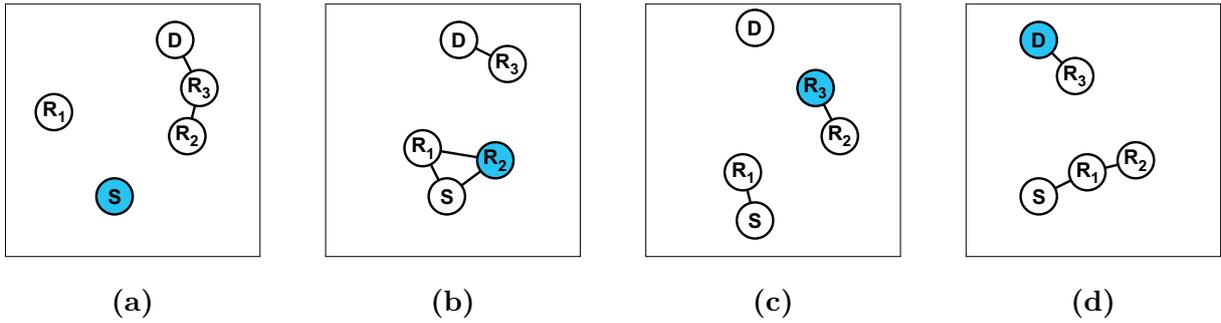
The world's first wireless network was the ALOHNET, which was developed at the University of Hawaii and became operational in June 1971 [1, 2]. A temporary experimental license was obtained in order to use two 100 kHz bandwidth channels at 407.350 MHz and 413.475 MHz, since frequency assignments for commercial applications were not available at that time. In May 1985, the Federal Communications Commission (FCC) established a set of rules for governing unlicensed access to the 902-928 MHz, 2400-2483.5 MHz, and 5725-5875 MHz radio bands, also known as Industrial, Scientific, and Medical (ISM) bands.

The first workshop on Wireless Local Area Networks (WLANs) was organized by the Institute of Electrical and Electronic Engineers (IEEE) in May 1991. At that time, the first WLAN products had just become available in the market, but without a common wireless standard they were rarely compatible. The first version of the IEEE 802.11 standard was released in 1997 [3] and since then several amendments have followed [4]. In 1999, a group of major companies formed the Wi-Fi Alliance, originally known as the Wireless Ethernet Compatibility Alliance (WECA), in order to promote WLAN products based on the IEEE 802.11 standard and to certify their interoperability. Nowadays, WLANs are widely used to extend existing wired networks in homes and offices, as well as to provide wireless Internet access in public places such as airports, restaurants, hotels, hospitals, and libraries.

The IEEE 802.11 standard defines two basic modes of operation, the infrastructure mode and the ad hoc mode. In infrastructure mode, mobile units communicate through access points that serve as gateways to an infrastructure network. Each access point, together with all the mobile units associated with it, form a basic service set (BSS). An extended service set (ESS) is a set of BSSs, connected by a distribution system. In ad hoc mode, mobile units communicate directly with each other, forming an independent basic service set (IBSS). Infrastructure mode may provide more stability, however the mobility is limited within the communication range of the access points. On the other hand, ad hoc mode offers greater mobility and flexibility, but most importantly it can achieve communication in circumstances that would not be possible with infrastructure mode, since it does not rely on any infrastructure.

The concept of Mobile Ad hoc Networks (MANETs) was introduced by the Defense Advanced Research Projects Agency (DARPA) with the Packet Radio Network (PRNet) project in 1973 [5, 6]. MANETs extend the ad hoc mode to provide multi-hop communication, which means that a mobile unit may serve as a router for others. This approach enhances the communication capabilities of the network, because a pair of mobile units can communicate even if they will never be within each other's communication range. The communication links between the mobile units of the network may vary over time due to their mobility, which means that the topology of the network is constantly changing. There are plenty of routing protocols that cope with the dynamic topology of these networks, such as DSDV [7], DSR [8], AODV [9], LAR [10], ZHLS [11], OLSR [12], and others [13, 14]. MANETs can be used for emergency networking in case of natural or man-made disasters, data collection with sensors, military operations, commercial and educational applications, coverage extension, and many more [15].

All the routing protocols that have been proposed for MANETs assume that there is always a contemporaneous end-to-end path for each pair of mobile units in the network. Unfortunately, this assumption may not be true in networks with frequent and long-lasting disconnections. In Intermittently Connected Networks (ICNs), also known as Challenged Networks, there is no guarantee that a fully connected path between any pair of mobile units exists at any time. Delay-Tolerant Networks (DTNs), also known as Disruption-Tolerant Networks, are applications in ICNs that tolerate delays beyond conventional forwarding [16]. The Delay-Tolerant Networking Research Group (DTNRG)



**Figure 1.1:** An example of data delivery in an Intermittently Connected Network

of the Internet Research Task Force (IRTF) has proposed a DTN architecture, which is based on an abstraction of message switching [17]. The provision of connectivity to remote regions [18, 19, 20], vehicular networks [21, 22, 23], wildlife tracking [24, 25], sparse sensor networks [26], military networks [27], and the Interplanetary Internet [28] are a few examples of such networks.

The continuous disconnections may complicate the task of data delivery in an ICN, but this problem can be reduced by taking advantage of the mobility of the nodes [29]. This approach led to the use of the store-carry-and-forward paradigm, in which a node receives and carries data from other nodes until a more suitable carrier is available. The routing protocols that have been proposed for these networks use various techniques to tackle this problem by exploiting the available information in the network.<sup>1</sup> Their purpose is to determine if an encountered node is suitable to carry a message, so that it will eventually reach its destination.

An illustration of the store-carry-and-forward paradigm is shown in Figure 1.1, as a series of snapshots from an ICN. In the first snapshot, node  $S$  generates some data that it wants to send to node  $D$ . However, since there is no one in its communication range, node  $S$  carries that data and waits for an opportunity to arise. In the second snapshot, node  $R_1$  and node  $R_2$  are within the communication range of node  $S$ . Then, based on the underlying routing algorithm, node  $S$  decides to forward the data destined for node  $D$  to node  $R_2$ . Similarly in the third snapshot, node  $R_2$  forwards the data to node  $R_3$ , and finally in the fourth snapshot, node  $R_3$  delivers the data generated from node  $S$  to node  $D$ . Although there was no contemporaneous path from node  $S$  to node  $D$  at any time, the data were successfully delivered by using the store-carry-and-forward paradigm.

<sup>1</sup>Some of the most prominent routing protocols are described in Section 2.1.

Furthermore, ICNs can be classified based on the knowledge of the topology. If the topology is deterministic and known, or at least predictable, the required forwards can be scheduled ahead of time [30]. In Opportunistic Networks the topology is stochastic and unknown, which leads nodes to seize the opportunities of forwarding data to other nodes. Since human mobility is often unpredictable, an Opportunistic Network can be formed by people carrying mobile devices that can communicate with each other. Hence, social network analysis techniques could be used in order to distribute data, since people tend to form communities. The Huggle project is mainly focused on prototyping this type of networks, which they refer to as Pocket Switched Networks (PSNs) [31].

## 1.2 Scope of the Thesis

The majority of the routing protocols that have been proposed for Opportunistic Networks uses some utility functions to determine how capable each node is to deliver a message to its destination. The nodes rely on these utility functions in order to route their messages. However, the fact that some nodes are more important than others, which corresponds to higher utility values, results to an unbalanced traffic load distribution that intensifies when the utility functions are destination-independent [32]. Even worse, the performance of the network degrades significantly due to the limited resources that cause congestion in the most important nodes. The main contributions of this thesis are the following:

1. We propose a congestion control mechanism that handles each message differently, based on the utility values and the saturation state of the relay nodes, which can be implemented in any utility-based routing protocol. Moreover, our congestion control mechanism provides a trade-off between efficiency and overhead by adjusting the fairness in the network.
2. We show that a significant performance gain can be obtained with a limited cutback in fairness. Thereby, we take advantage of the most important nodes in the network without overusing them, while maintaining a relatively fair resource allocation. We evaluated several congestion control mechanisms on three different real-world data sets, from which similar conclusions were drawn.

## 1.3 Overview of the Thesis

The rest of the thesis is organized as follows. Chapter 2 reviews state-of-the-art routing protocols that have been proposed for Opportunistic Networks and strategies to cope with the resource constraints. Chapter 3 demonstrates the need for adjustable fairness in these networks and describes in detail the proposed congestion control mechanism. Chapter 4 describes the simulation environment and the evaluation metrics that were used, followed by the results of the simulations. Chapter 5 summarizes our findings and provides a list of possible extensions of this work. Finally, detailed descriptions of the two main routing protocols that were used in this thesis are given in Appendices A and B respectively, while a further experimental analysis of our congestion control mechanism can be found in Appendix C.

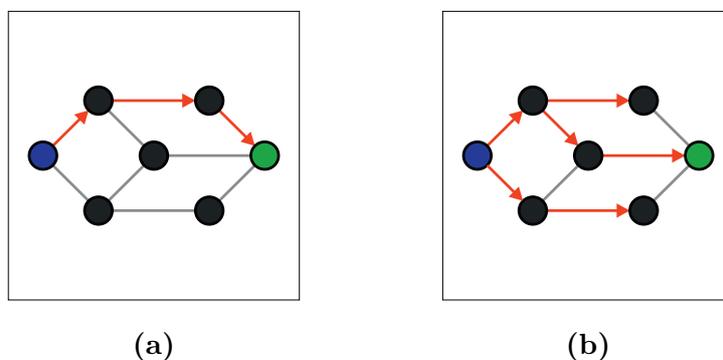
# Chapter 2

## Background and Related Work

### 2.1 Routing in Opportunistic Networks

Each routing protocol that has been proposed for Opportunistic Networks can be classified as either single-copy [33] or multi-copy [34], regardless of the heuristic methods that it may use. In single-copy routing protocols, each message is unique in the network. Therefore, when a message is forwarded to a new carrier, the previous one deletes it. On the other hand, in multi-copy routing protocols, each message may have multiple copies carried by multiple nodes in the network. However, the process of replication can be defined variously for each routing protocol.

Figure 2.1 depicts the differences in the process of delivering a message with a single-copy routing protocol and a multi-copy routing protocol. It is clear that the single-copy approach performs fewer transmissions and consumes less storage resources than the multi-copy approach. However, since the multi-copy approach spreads each message



**Figure 2.1:** An illustration of (a) Single-Copy and (b) Multi-Copy Routing

across multiple nodes, it has a higher probability of successful delivery and it may also be able to deliver the message faster than the single-copy approach. Eventually, the choice between single-copy and multi-copy routing is a trade-off between performance and overhead.

Flooding-based routing protocols are by definition multi-copy routing protocols, since each node forwards a copy of each message that it holds to all or part of the nodes that it encounters. Utility-based routing protocols use some utility functions to make their routing decisions, which means that they can operate as either single-copy or multi-copy routing protocols. These utility functions can be destination-dependent, destination-independent, or a combination of both. However, there are several types of utility-based routing protocols that differ on the techniques that they use. History-based routing protocols use the history of encounters to estimate their likelihood to be repeated. Social-based routing protocols attempt to predict future interactions by exploiting the social behavior of the nodes.

An outline of the most notable routing protocols for each of these categories is given below. It should be noted that there are several other categories of routing protocols for Opportunistic Networks that rely on different techniques. For instance, some routing protocols exploit temporary network partitions to use MANET techniques [35, 36], while others may even use infrastructure devices or mobility characteristics in order to enhance the performance of the network [37, 38].

### **2.1.1 Flooding-Based Routing Protocols**

One of the earliest approaches to cope with the intermittent connectivity of these networks is Epidemic Routing [39]. Each node maintains a summary vector that indicates the messages that it carries. When two nodes meet, they exchange their summary vectors and each node requests copies of the unknown messages that the other node carries. In scenarios without tight resource constraints, Epidemic Routing is able to deliver most of the messages, by spreading them to all the nodes in the network and eventually reaching their destinations. However, Epidemic Routing not only has high overhead due to the many unnecessary transmissions, but the performance may also significantly degrade if the resources are limited [40, 41].

Several routing protocols have been proposed that aim to control the flooding of messages in the network [42]. The Spray and Wait routing protocol [43] reduces the overhead of flooding by bounding the number of copies each message has in the network. The routing process is divided into two phases, the spray phase and the wait phase. In the spray phase, a fixed number of copies of the original message is disseminated in the network. In the wait phase, each node that carries a copy of the message will wait to meet its destination to deliver it directly. Afterwards, in the Spray and Focus routing protocol [44], the wait phase was replaced with the focus phase. In the focus phase, each node can forward the copy that it holds to another node, based on a utility function. Network coding can also be used to reduce the number of transmissions in the network [45, 46, 47, 48]. Instead of simply transmitting one packet at a time, each node can combine several packets and transmit them as one.

### 2.1.2 History-Based Routing Protocols

The most representative history-based routing protocol is P<sub>Ro</sub>PHET, which stands for Probabilistic Routing Protocol using History of Encounters and Transitivity [49, 50]. This approach is based on the assumption that if two nodes have met several times in the past, it is very likely that they will meet again in the future. Each node maintains a delivery predictability for each known destination, which indicates how likely this node is to encounter a certain destination in the future. When two nodes meet, they exchange their delivery predictabilities so that they can update their estimates. The nodes that are frequently encountered have high delivery predictabilities for each other that decrease over time to avoid obsolete information. A transitive property is also applied to these estimates, so that the messages can be forwarded to nodes that encounter their destinations indirectly. Therefore, each message is forwarded to the node that has the highest delivery predictability for its destination. An improved version has also been proposed, called P<sub>Ro</sub>PHETv2 [51], where the equations that update the delivery predictabilities have been refined to solve two problems that were later observed.

Similarly, in a routing protocol called NECTAR [52], each node calculates a Neighborhood Index for all the other nodes in the network based on the history of its contacts. This index is then used to determine the route that each message should take, since it

favors the nodes that have met the destination recently. MEED is another history-based routing protocol, which stands for Minimum Estimated Expected Delay [53]. Each node records the contact and inter-contact duration that it has with the other nodes in the network over a sliding history window, in order to estimate the expected time for the next contact.

### 2.1.3 Social-Based Routing Protocols

Social-based routing protocols take advantage of the long-term social relationships between the users of an Opportunistic Network in order to achieve data delivery. A wide range of human mobility traces have been analyzed to develop realistic models of human mobility and to determine the extent of its predictability [54, 55], as well as to study the characteristics of the contacts between human-carried devices [56, 57, 58]. It has been observed that in most real social networks, each pair of individuals is connected through a short chain of acquaintances, which is known as the small-world phenomenon [59, 60]. It has also been shown that the physical encounters of mobile devices carried by people are sufficient to build social graphs with small-world properties [61]. The combination of short average path lengths and high clustering coefficient of their structure makes them suitable for routing decisions [62, 63, 64]. Numerous social measures have been proposed in order to evaluate the significance of each node in the social graph and to predict future interactions [65, 66, 67, 68].

SimBet [69] uses two social metrics, betweenness centrality and similarity, on a social graph that constructs based on the contacts of the nodes. The betweenness centrality metric is used to identify “bridges” in the social graph, while the similarity metric is used to identify nodes that are “close” to the destination. Unfortunately, the betweenness centrality measure cannot be used in real networks, because it requires complete knowledge of the network. Alternatively, each node calculates the betweenness centrality of its ego network, called egocentric betweenness [70], that consists of the ego node, its neighbors, and all the links among them.<sup>1</sup> An extension of SimBet was later proposed, named SimBetTS [71], which includes another social metric, called Tie Strength, in order to evaluate the “strength” of each link in the social graph.

---

<sup>1</sup>Further details on the SimBet algorithm are given in Appendix A.

Bubble Rap [72, 73] relies on the concept of community and centrality in order to solve the routing task. It is assumed that each node belongs to at least one community, even if it is a single-node community. Additionally, each node has a global centrality across the whole network and a local centrality for each of the communities that it belongs. If a node carries a message destined for a node that belongs in another community, it uses the global centrality to forward the message, until a node that belongs to the same community with the destination is found. Then, the forwarding decisions are made based on the local centrality of the community, until the destination is reached. Therefore, Bubble Rap requires a community detection mechanism, as well as a technique to approximate the centrality values [74, 75, 76].

Friendship Based Routing [77, 78] has also been proposed, that exploits the concept of friendship in order to make routing decisions. The quality of friendship of two nodes is defined by the time that they were connected over a certain period. Then, friendship communities of nodes are constructed, that have a quality of friendship greater than a threshold. Messages are forwarded only to nodes that are stronger friends than the current carrier and in the same friendship community with their destinations.

## 2.2 Coping with Resource Constraints

Regardless of the routing heuristics that may be used in an Opportunistic Network, there are several resource limitations that may hinder the successful delivery of messages. Since the nodes have buffers of finite capacity, they can only carry a certain amount of messages at a time. In addition to that, we also have to consider about the limited bandwidth, the limited computing power, and the energy consumption of each node.

Efficient scheduling and drop policies are essential for the performance of the network. Scheduling policies are used to determine the importance of each message, so as to give priority to the most important messages, because limited bandwidth and unexpected contact interruptions may prevent the transmission of all the anticipated messages. Drop policies are used to determine which message to discard when congestion occurs. Some of these policies use local information such as the number of times each message has been forwarded [79, 80], while others use network-wide information such as the number of copies each message has in the network [81, 82].

RAPID [83, 84] is one of the first routing protocols that takes into account resource constraints and handles the routing process as a resource allocation problem. The order in which the messages are replicated depends on their utilities, so that a specific routing metric can be optimized. Similarly, ORWAR [85] replicates the messages with the highest utility values first and discards the messages with lowest utility values when needed. Additionally, Delegation Forwarding [86] has been proposed that reduces the overhead of transmissions by forwarding messages only to the highest-quality nodes.

Several congestion control strategies have been proposed for Opportunistic Networks that aim to maintain high delivery ratio and low delay, as if there were no resource constraints [87]. These strategies can be classified based on the number of copies that the underlying routing protocol uses to deliver each message. Some of the most notable strategies are described below.

### 2.2.1 Strategies for Single-Copy Routing Protocols

In single-copy routing protocols, the packets that each node is forced to drop will not be able to be delivered to their destinations, since there are no other copies in the whole network. Therefore, some packets may be delivered from alternative paths that are less congested, in order to avoid as many packet drops as possible. However, the use of alternative paths may lead to an increase in delivery time [88].

FairRoute [89] is a routing protocol that uses a queue control to distribute the traffic load fairly among the nodes of the network. It relies on the interaction strength at different time scales, but it also examines the queue size of each node in order to make routing decisions. FairRoute uses the queue length of each node as an equivalent of its social status, so that each node will receive messages only from nodes of equal or higher status.<sup>2</sup> Since the queue control mechanism is independent from the routing mechanism, it can also be implemented in other routing protocols to increase their fairness.

Economic and financial models have also been proposed for congestion control in Opportunistic Networks, so that each node can decide autonomously whether to accept to carry a message or not [90, 91]. Another approach that has been introduced for Congestion Aware Forwarding, called CAFé [92], utilizes buffer and network statistics to

---

<sup>2</sup>The detailed description of the FairRoute algorithm can be found in Appendix B.

predict if the transmission of a message will later cause congestion. Alternatively, Storage Routing [93, 94] relies on the mitigation of messages from the congested nodes to their neighbors.

### 2.2.2 Strategies for Multi-Copy Routing Protocols

Dynamic replication management is needed in multi-copy routing protocols in order to deal with the problem of storage congestion. In this case, we can tolerate packet drops to some extent, because there are multiple copies of each packet in the network. Consequently, the buffers of some nodes may be overfilled a lot faster due to the increased traffic load.

A variety of methods that use “anti-packets” have been proposed, whose aim is to make better use of the storage capacity of each node by erasing redundant replicas [95]. According to the VACCINE method, when a node delivers a packet to its destination successfully, it erases its copy of the packet and it keeps an identifier of the delivered packet, which corresponds to an “anti-packet”. Whenever two nodes meet, they exchange their “anti-packets” in order to erase the packets that are no longer needed and to prevent from receiving them again in the future.

Another approach that has been proposed for congestion control in multi-copy routing protocols aims to control the replication rate, in accordance with the level of congestion in the network [96]. Similarly, CAFREP [97] controls the number of copies that each node forwards to another node, according to their buffer and network statistics. Thereby, the replication rate changes over time based on the available resources.

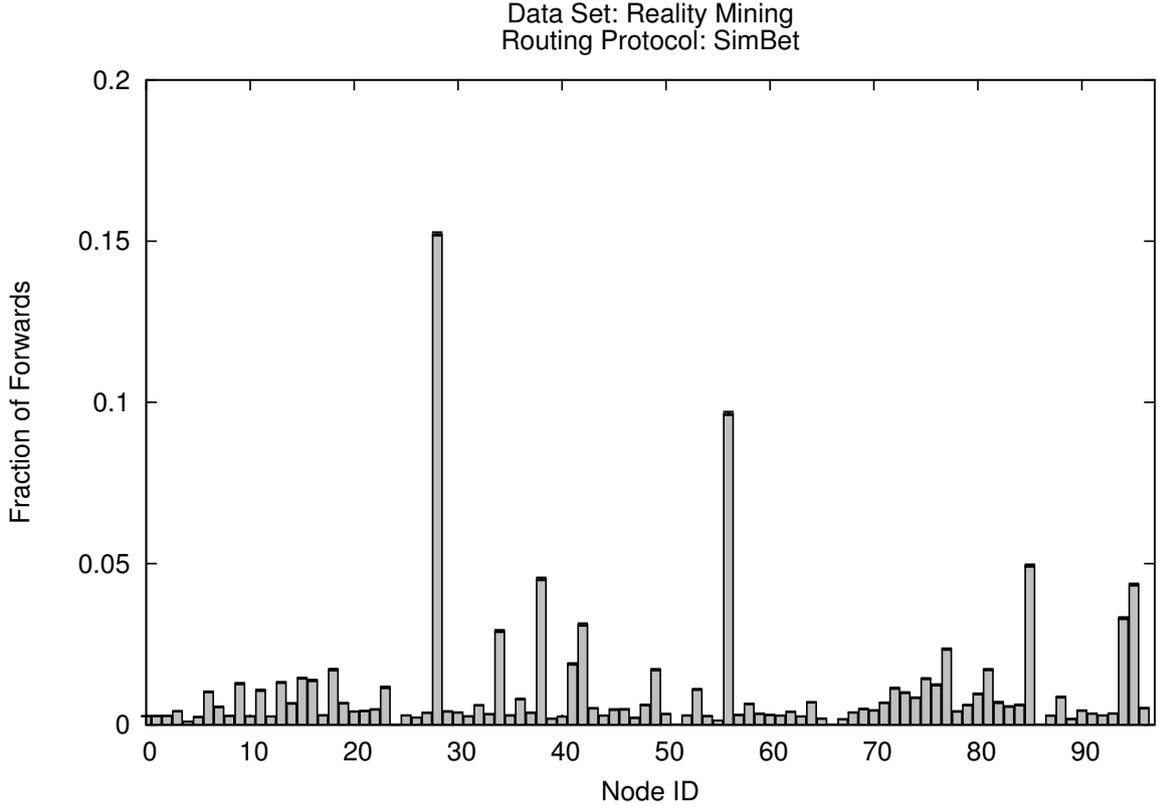
# Chapter 3

## Congestion Control in Opportunistic Networks

### 3.1 The Need for Adjustable Fairness

Most of the routing protocols that have been proposed for Opportunistic Networks make greedy routing decisions that aim to maximize a certain utility function, regardless of any resource constraints. When two nodes meet, each message is simply forwarded to the node with the highest utility value for its destination. These utility values are used to determine which node is more likely to deliver a message to its destination. However, this process does not take into account storage, bandwidth, or energy constraints. For instance, according to the SimBet [69] routing protocol, each message is forwarded to the node with the highest SimBet utility value, even if its buffer is already fully occupied. More specifically, node  $i$  will forward a message, destined for node  $d$ , to node  $j$  if it is true that  $SimBetUtil_j(d) > SimBetUtil_i(d)$ .

This approach leads to an unbalanced traffic load distribution in the network, since most of the packets are forwarded to the nodes with the highest utility values, which is even more evident when the utility functions are destination-independent [32]. Due to storage limitations, these nodes are most of the time under congestion, resulting in loss of messages. Even worse, in the case of single-copy routing, each time a node is forced to drop a packet, it is permanently lost because there is no other copy in the network. Likewise, bandwidth and battery limitations would affect the performance dramatically,



**Figure 3.1:** Distribution of Forwards without resource constraints

since the nodes with the highest utility values perform most of the transmissions. On the other hand, the nodes with the lowest utility values perform far fewer transmissions, since they are rarely used as intermediate nodes. This phenomenon is particularly aggravated in the case of social-based routing protocols, because most of the traffic load is carried by the most popular nodes.

Figure 3.1 shows the distribution of forwards, from simulations of SimBet using the human mobility traces from the MIT Reality Mining data set [98], without any resource constraints.<sup>1</sup> It is clear that a small subset of the nodes account for most of the forwards that occurred in the whole network. From a total of 97 nodes, over 50% of the total number of forwards was held just by the top 6 nodes. If there were storage constraints, most of these packets would be dropped due to congestion, decreasing the efficiency of the routing protocol significantly. Even if the top 6 nodes could carry all of these packets, their batteries would get drained a lot faster. Similar results have been observed for other routing protocols as well, in a variety of data sets [72, 86, 99, 100].

<sup>1</sup>More details about the simulation setup will be given in Section 4.1.

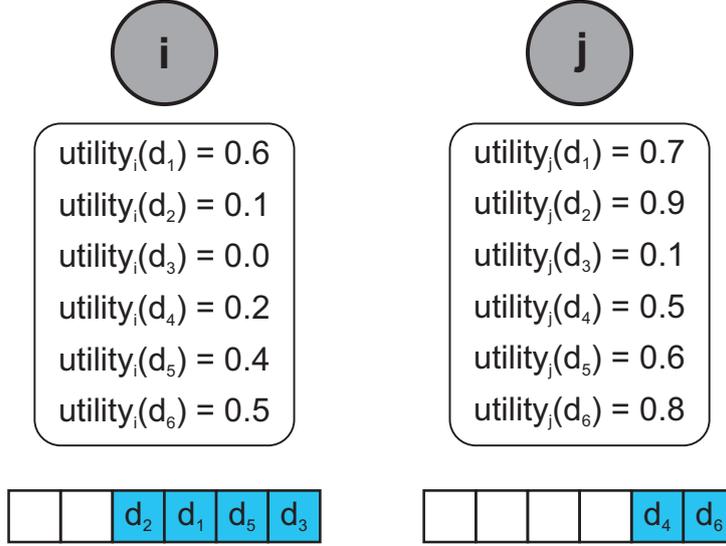
A well-known solution to the unbalanced traffic load distribution is to increase the fairness in the network. As a result, the traffic load is distributed fairly among all the nodes in the network, which leads to a reduction in the total number of packet drops. Unfortunately, if we distribute the packets with absolute fairness, we lose in terms of delivery ratio and delay [101]. The reason why this happens is because most of the opportunities to forward a packet to a better node will be lost, just to keep the fair allocation of resources. Even though there may be several alternative paths to reach the destination, the delivery time may increase significantly [88]. On the other hand, by forwarding packets to the nodes with the highest utility values, we increase the likelihood of delivering them to their destinations and thus reducing the traffic load in the network, which corresponds to fewer packet drops. Therefore, our goal is to be as fair as possible, while taking advantage of the important nodes wisely.

## 3.2 Congestion Control with Adjustable Fairness

In this section we delineate the proposed congestion control mechanism, as well as the intuition behind it. Initially, we present our methodology for the measurement of the importance of each packet transmission. Then, we describe how our congestion control strategy differs from other known approaches. Finally, we define the forwarding condition of our congestion control mechanism, which also provides a trade-off between efficiency and fairness through a tunable parameter.

First of all, we need to be able to determine the importance of each forwarding step. Obviously, when the receiver has much higher utility value than the sender, the likelihood of successful delivery is significantly increased. However, we also have to take into account the cases where their utility values are close to 0. The most unfavorable situation is when a node carries packets for which it has a utility value equal to 0. In such occasions, it is crucial to forward these packets to nodes that have at least a little higher utility values, so that there will be a greater chance to find other nodes that will be able to deliver them to their destinations.

For that purpose, we use the normalized utility value  $U_{i,j}(d)$  that we define in Equation 3.1, where  $i$  and  $j$  are the two encountered nodes and  $d$  is the destination of a packet. This normalized utility value is equal to 1 only when the current carrier has a utility value that

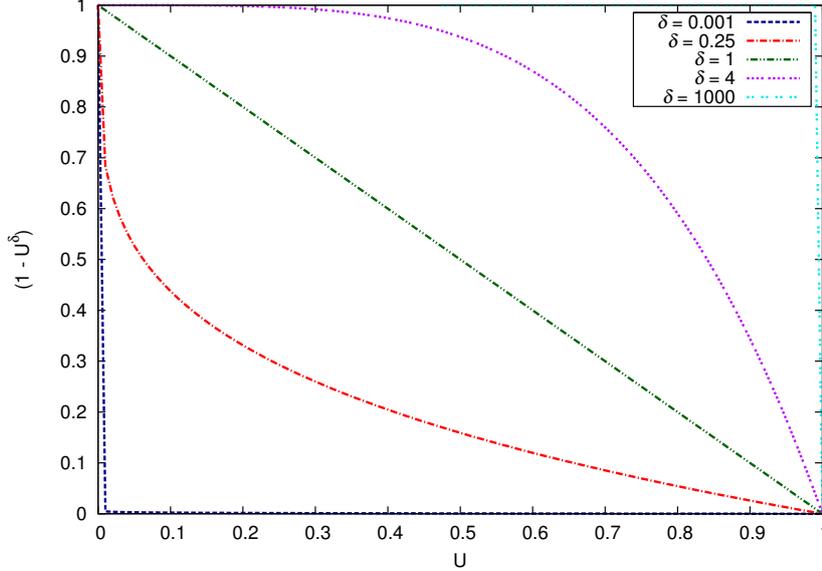


**Figure 3.2:** Two nodes in contact with known utility values and buffer contents

it is equal to 0 and it encounters another node with a utility value greater than 0. However, this normalized utility value is also close to 1 when the encountered node has a much higher utility value than the current carrier, which means that it is much closer to the destination. On the contrary, when the normalized utility value is close to 0, it indicates that both nodes have almost equal utility values. This normalized utility value can be used in any utility-based routing protocol, without affecting its performance. For instance, in the SimBet routing protocol, the forwarding condition  $SimBetUtil_j(d) > SimBetUtil_i(d)$  is equivalent to  $U_{i,j}(d) > 0$ .

$$U_{i,j}(d) = \frac{utility_j(d) - utility_i(d)}{utility_j(d) + utility_i(d)} \quad (3.1)$$

Figure 3.2 depicts a scenario with two nodes *i* and *j* carrying some packets for the destination nodes  $d_1, \dots, d_6$ . Without any congestion control mechanism, node *i* would forward all the packets that it carries to node *j*, simply because node *j* has a higher utility value for each one of them. According to other proposed strategies, such as FairRoute [89], node *i* would keep forwarding packets to node *j* until the traffic load is distributed fairly. In our approach, we may violate the fair traffic load distribution to forward some crucial packets, based on the normalized utility value  $U_{i,j}(d)$ . For example, we would not forward the packet destined for node  $d_1$ , because both nodes have almost equal utility values, but we may forward the packet destined for node  $d_2$ , since node *j* has a much higher utility value than node *i*.



**Figure 3.3:** The impact of  $\delta$  on the forwarding condition

Therefore, we also need a measure of the saturation state of each node. We use the remaining storage space so that when two nodes meet, they both know how many more packets each node can carry, avoiding unnecessary packet drops. However, since there is no guarantee that all nodes can carry the same total amount of data, we use the normalized residual space. The normalized residual space of node  $i$  is calculated by dividing its remaining storage space  $R_i$  with the total amount of storage space  $B_i$  that it can allocate. Equation 3.2 defines the forwarding condition of the proposed congestion control mechanism, which has to be true in order to forward a packet from node  $i$  to node  $j$  that is destined for node  $d$ .

$$(U_{i,j}(d) > 0) \wedge \left( \frac{R_j}{B_j} > (1 - (U_{i,j}(d))^\delta) \frac{R_i}{B_i} \right) \quad (3.2)$$

We also introduce a tunable parameter  $\delta$ , so that we can adjust our approach either towards high efficiency or absolute fairness. By raising the normalized utility value  $U_{i,j}(d)$  to the power of  $\delta$ , we can choose an appropriate value to define it so that the performance of the network can meet the needs of a certain application. Figure 3.3 demonstrates the impact that five different values of  $\delta$  have on the forwarding condition.<sup>2</sup> When we assign a value to  $\delta$  from the interval  $(0, 1)$ , we allow the transmission of even more packets with a high normalized utility value  $U_{i,j}(d)$  by relaxing the forwarding condition. On the other

<sup>2</sup>We examine the impact that these values have on the performance of the network in Appendix C.

hand, by assigning a value to  $\delta$  greater than 1, we mostly concentrate on maintaining the load balancing among the nodes by enforcing the forwarding condition. Therefore, we can fine-tune the  $\delta$  parameter to achieve either low end-to-end delay or low overhead.

To summarize, in the case where  $U_{i,j}(d) \rightarrow 0$ , the packet will be forwarded to node  $j$  only if it also contributes at balancing the traffic load between the two nodes. When  $U_{i,j}(d) \rightarrow 1$ , we may break the load balancing between their buffers in order to forward the packet to node  $j$  and increase the likelihood of successful delivery. All the packets for which it is true that  $U_{i,j}(d) = 1$  will be forwarded to node  $j$ , as long as it has enough space to store them, which means that  $R_j$  must be greater than or equal to the size of the corresponding packet. It should be noted that when  $\delta \rightarrow \infty$ , our approach resembles the queue control of FairRoute [89] and when  $\delta \rightarrow 0$  we operate as if there was no congestion control mechanism, but without overfilling the buffers of the nodes.

### 3.3 Exploiting Social Preferences and Heterogeneity

In social psychology, the attitude that each individual has for the gain of others in relation to their own is often called Social Value Orientation (SVO), with several different methods for its measurement [102, 103, 104, 105]. Selfish people show no concern about the needs of others but themselves, while altruistic people willingly sacrifice their own welfare for the benefit of others. The  $\delta$  parameter can be considered as the social preferences that each node has towards allocating its resources for the sake of others. When  $\delta$  is close to 0 the nodes behave altruistically, because they are contributing all of their resources to help others achieve communication. Selfish nodes would have high  $\delta$  values, so that they could avoid using their resources to help others.

In this work, we assume that the  $\delta$  parameter is the same fixed number for all the nodes in the network. Therefore, in order to achieve high delivery ratio and low delay, we need to define the  $\delta$  value so that the nodes will behave cooperatively. However, each node could define its own  $\delta$  value in accordance with its social preferences, which could also change over time. In that case, the most efficient solution may result from a mixture of selfish, prosocial, and altruistic nodes. It is evident that a self-configuring method is required in order to dynamically adjust the value of the  $\delta$  parameter, which we intend to investigate in future work.

Furthermore, in a real scenario it is very likely that the total amount of storage space would vary from node to node. The performance of the network would be significantly affected if some of the most important nodes had very limited storage space. Even worse, if these nodes were also acting selfishly, it would require much more time to deliver each packet through alternative routes. Additionally, in cases where there are energy constraints, we should also take into account the battery level of each device in order to avoid node overloading.

# Chapter 4

## Experimental Evaluation

### 4.1 Simulation Setup

In collaboration with the other members of the Networks Research Group, we have developed a custom simulator in order to evaluate the performance of various routing protocols that have been proposed for Opportunistic Networks. We have implemented plenty of routing protocols that have been described in the literature, as well as strategies to cope with resource constraints, with a variety of mobility traces available for simulations. Table 4.1 summarizes the characteristics of the data sets that are being used throughout this thesis, which are available in the CRAWDAD archive [106].

**Table 4.1:** Characteristics of the data sets

<b>Data Set</b>	<b>Reality Mining</b>	<b>PMTR</b>	<b>Sassy</b>
<b>Institution</b>	Massachusetts Institute of Technology	University of Milan	University of St Andrews
<b>Number of Nodes</b>	97	44	25
<b>Duration</b>	290 days	19 days	74 days
<b>Radio Range</b>	10 meters	10 meters	10 meters
<b>Granularity</b>	300 seconds	1 second	6.67 seconds
<b>Number of Contacts</b>	113875	11895	112264

**Table 4.2:** Simulation Settings

<b>Traffic Type</b>	Uniform
<b>Warm-Up Period</b>	20% of Simulation Time
<b>Cool-Down Period</b>	20% of Simulation Time
<b>Buffer Size</b>	20 packets
<b>Scheduling Policy</b>	FIFO
<b>Drop Policy</b>	Drop Front
<b>Bandwidth</b>	Infinite
<b>TTL</b>	Infinite
<b>Number of Trials</b>	25

The first data set, called Reality Mining [98], consists of contacts from 97 students and faculty members at the MIT. They carried mobile phones that were performing Bluetooth scans every 5 minutes. The second data set, called PMTR [107], contains contacts from 44 students, faculty members, and technical staff at the University of Milan. They used custom-made devices that were designed to perform scans on a per-second basis. The third data set, called Sassy [108], comprises contacts from 25 sensors that were carried by individuals associated with the University of St Andrews. The sensors attempted to detect each other every 6.67 seconds. Each of these data sets has its own advantages and disadvantages due to their different number of participants, durations and scanning intervals.

Table 4.2 provides the simulation settings of the experiments that we performed. The traffic of each simulation was generated uniformly at random, so that each packet was created at a random time, with a random pair of source and destination nodes. To avoid statistical bias, the results were collected only after a warm-up period and before a cool-down period, each of which lasts as much as 20% of the total simulation time, so that the network would be in its steady state. Each node could store up to 20 packets in its buffer, that processes in a FIFO and Drop Front order. We assumed that there are no bandwidth, time-to-live or energy limitations, so that the storage space would be the only constraint of our experiments. We examined six different scenarios, where the traffic load

ranges from low to high, based on the characteristics of each data set. For each scenario we simulated 25 trials in order to calculate the average values and the 95% confidence intervals of a variety of evaluation metrics.

## 4.2 Evaluation Criteria

In order to evaluate the performance of each congestion control strategy, we have to define a set of evaluation metrics. For each simulation trial, we collected the required data for the calculation of these evaluation metrics. The definition of each evaluation metric is given below.

### Delivery Ratio

The delivery ratio is defined as the total number of packets that were successfully delivered, divided by the total number of packets that were generated. In Opportunistic Networks, the delivery ratio is rarely close to 100%, mainly because of the intermittent connectivity of the nodes, the unknown topology, and the resource constraints. Even without these limitations, some packets cannot be delivered due to the absence of a sequence of intermediate nodes that can connect the source node with the destination node at any time.

### Average Delay

Even though the applications in Opportunistic Networks must be able to tolerate long delivery times, there are many applications that could benefit from delivering the packets in the minimum possible time. We calculate the average delay by adding all the delivery times and then dividing them by the total number of packets that were successfully delivered. Because the process of packet delivery often takes a long time, we measure the average delay in hours. However, the average delay takes into account only the packets that were successfully delivered, regardless of the total number of packets that the routing protocol failed to deliver. For that reason, we also use the cumulative distribution function of the delay, which describes the fraction of packets with a delivery time less than or equal to a certain time.

## Overhead

We use the total number of forwards that occurred in the whole network as an approximate measure of the energy consumption. In order to be able to compare the overhead between simulations with different traffic load, we normalize the total number of forwards by dividing them with the total number of packets that were generated. Additionally, we also calculate the cumulative fraction of forwards in order to evaluate the fairness of each algorithm, in terms of energy consumption.

## Average Number of Hops

The number of hops refers to the number of forwards that occurred in order to deliver a packet to its destination. The average number of hops is calculated by adding the number of hops of the delivered packets, divided by the total number of packets that were successfully delivered. In other words, this metric indicates the average length of the successful routing paths.

## Total Number of Packet Drops

Due to the constraints on storage capacity, a node may be forced to drop some of its packets, especially if the routing protocol does not use a congestion control mechanism. However, even if we use an optimal congestion control mechanism, if the resources of the nodes are not enough to support the overall traffic load, some packets will have to be dropped. Unfortunately, in the case of single-copy routing, each packet drop automatically reduces the delivery ratio, since there is no other copy of the packet in the network. In order to determine if congestion occurs only into a small subset of nodes, we also calculate the cumulative fraction of packet drops.

## 4.3 Performance Analysis

In this section we are going to compare the performance of four different algorithms on three different data sets. The forwarding condition of each of these algorithms is specified in Table 4.3, where  $U_{i,j}(d) = \frac{SimBetUtil_j(d) - SimBetUtil_i(d)}{SimBetUtil_j(d) + SimBetUtil_i(d)}$  since we use SimBet [69] as our reference routing protocol, so that we can compare the effectiveness of three

**Table 4.3:** The forwarding condition of each algorithm

Algorithm	Forwarding Condition
SimBet	$U_{i,j}(d) > 0$
SimBet-AF	$(U_{i,j}(d) > 0) \wedge \left( \frac{R_j}{B_j} > \left(1 - (U_{i,j}(d))^\delta\right) \frac{R_i}{B_i} \right)$
SimBet-FR	$(U_{i,j}(d) > 0) \wedge ((Q_j \leq Q_i) \vee (U_{i,j}(d) = 1))$
SimBet-LB	$(U_{i,j}(d) > 0) \wedge \left( \frac{R_j}{B_j} > \frac{R_i}{B_i} \right)$

different congestion control strategies. For every node  $i$  in the network,  $R_i$  corresponds to its remaining storage space,  $B_i$  corresponds to the total amount of storage space that it can allocate, and  $Q_i$  is equal to its queue length. The original version of SimBet forwards the packets greedily to the nodes with the highest utility values, regardless of their availability in storage resources. The original forwarding condition of SimBet is  $SimBetUtil_j(d) > SimBetUtil_i(d)$ , which is equivalent to  $U_{i,j}(d) > 0$ .

We implemented our proposed congestion control mechanism in SimBet, which we refer to as SimBet-AF. The major advantage of our approach compared to others is that it handles each packet differently, based on its importance and the saturation state of the nodes. It also enables the adjustment of fairness through a tunable parameter  $\delta$ , so that we can choose between high performance or low overhead. During the simulations from which we collected the following results, we had assigned  $\delta = 0.25$ , based on our empirical observations.<sup>1</sup> In future work, we are going to investigate methods to define the  $\delta$  value dynamically, based on local information.

To our knowledge, the most relevant work in the literature is the queue control of the FairRoute [89] routing protocol, in which each node will accept to carry a packet from another node, only if the other node has a longer or equal queue length. However, in case that the current carrier of a packet has a utility value equal to 0, it will be accepted regardless of their queue lengths, if the other node has a utility value greater than 0. We implemented the queue control of FairRoute in SimBet, which we call SimBet-FR, so that we can compare its performance with the other approaches.

---

<sup>1</sup>Our experiments in Appendix C show that the most efficient values of  $\delta$ , in terms of delivery ratio and delay, lie in the interval of  $(0, 1)$ .

In order to examine how beneficial the utility values are for a congestion control mechanism, we have also performed simulations for an algorithm that we refer to as SimBet-LB. This approach aims to distribute the load among the nodes with absolute fairness, regardless of the importance of each packet. According to the SimBet-LB algorithm, a node will accept to carry a packet from another node if and only if the sender is in a higher saturation state.

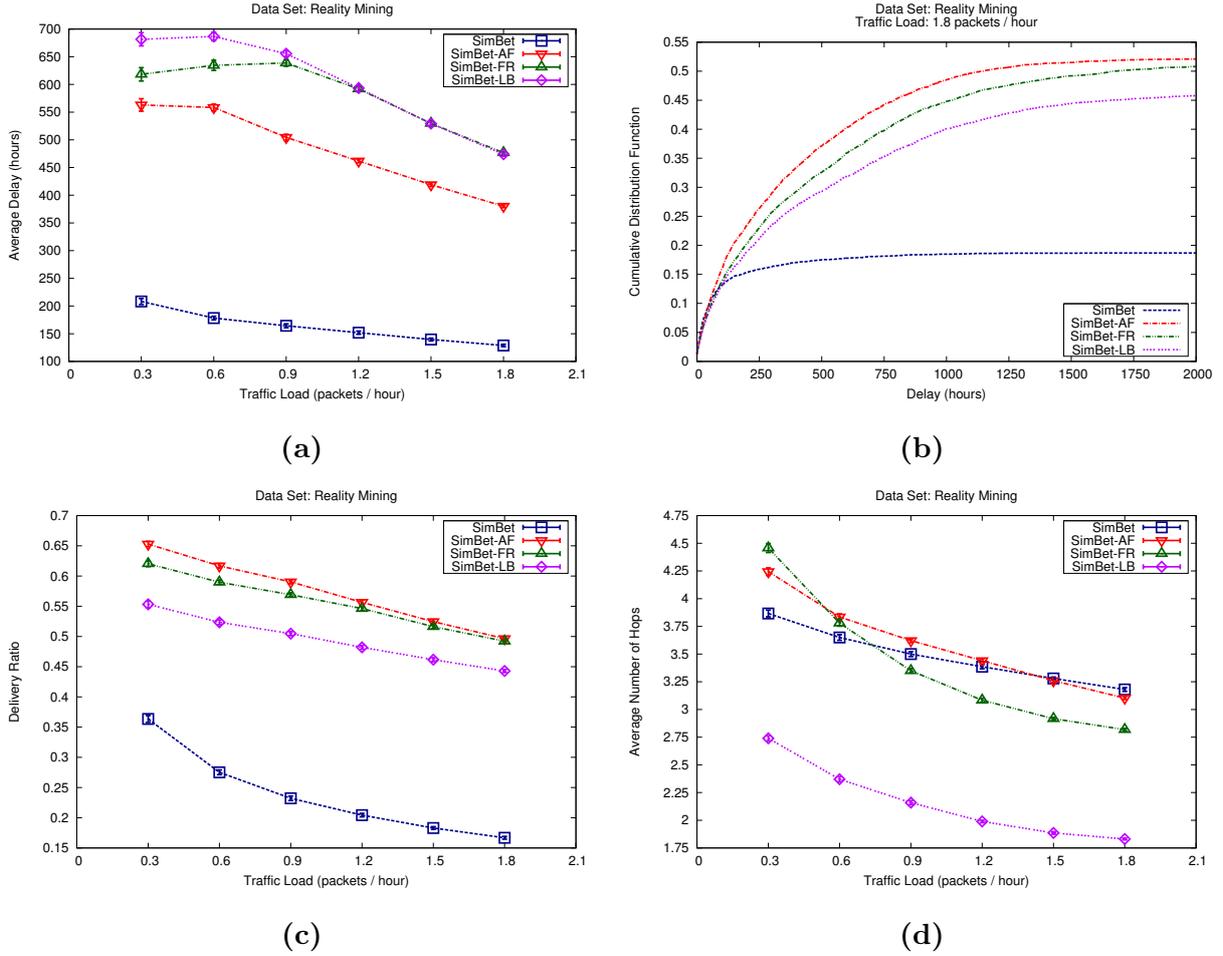
We performed simulations on three different data sets, which are known as the Reality Mining, the PMTR, and the Sassy data sets. Their characteristics were previously presented in Table 4.1. In subsection 4.3.1, we are going to analyze in detail the performance of each algorithm on the Reality Mining data set, which is the most widely used for simulations in such networks. Finally, in subsection 4.3.2 we analyze the results of the most important evaluation metrics from simulations on the PMTR and Sassy data sets.

### 4.3.1 Simulations on the Reality Mining Data Set

As we can observe in Figure 4.1a, from the three congestion control strategies that we implemented, SimBet-AF achieves the lowest average delay under any traffic load. The SimBet-LB algorithm has the highest delivery time on average, because it tries to balance the load among each pair of nodes of the network. It should be noted that on average, as the traffic load increases, the SimBet-FR algorithm delays to deliver the packets to their destinations as much as the SimBet-LB algorithm. Our approach reduces the delivery time significantly, because it may violate the load balancing in order to seize some of the most crucial forwarding opportunities.

Plain SimBet has the lowest average delay, because it can only deliver the packets that do not have to stay in the buffers of the intermediate nodes for a long period of time. However, as Figure 4.1b shows, the other three algorithms were able to deliver these packets in a short period of time as well. Even though SimBet-FR and SimBet-LB have almost the same average delay, SimBet-FR was able to deliver more packets. Nevertheless, SimBet-AF manages to deliver even more packets to their destinations, but also in a much shorter period of time.

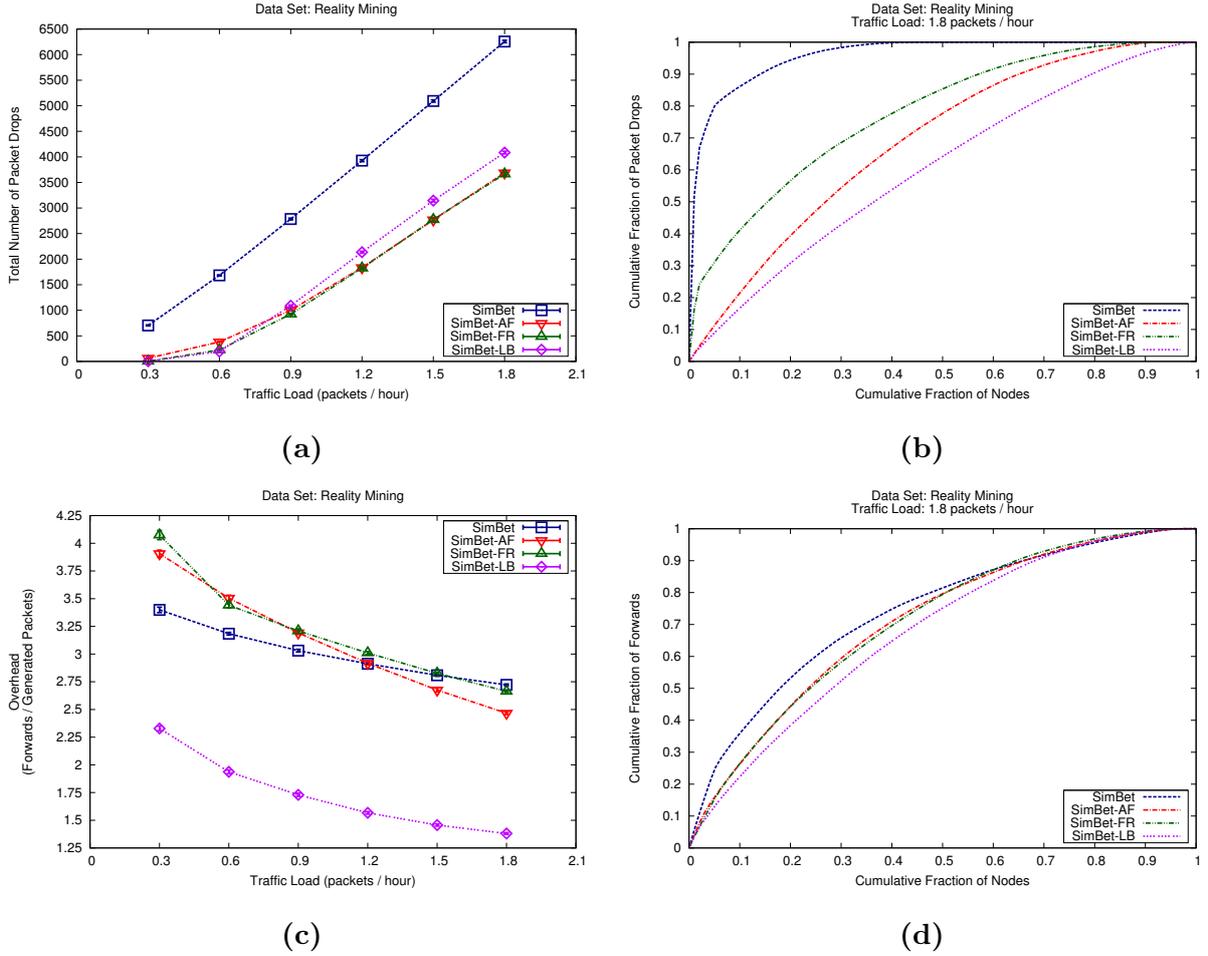
Figure 4.1c clearly demonstrates the impact of storage constraints on the performance of the network, as well as the need for congestion control. Plain SimBet could only



**Figure 4.1:** (a) Average Delay, (b) Cumulative Distribution Function of the Delay, (c) Delivery Ratio, and (d) Average Number of Hops on the Reality Mining data set

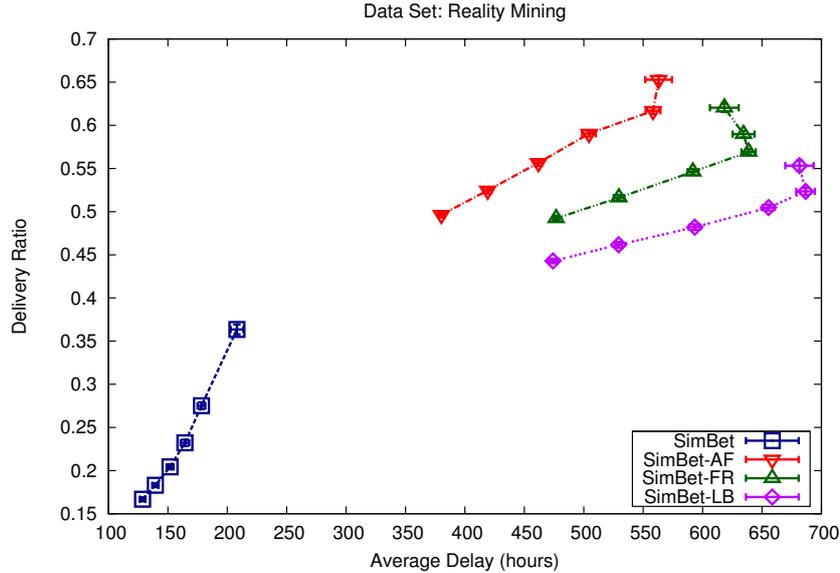
deliver about 35% of the packets that were generated under low traffic load, while it barely surpassed 15% under high traffic load. Even the SimBet-LB approach was able to achieve a higher delivery ratio, simply because it does not forward the packets blindly. As we can observe, under high traffic load, both SimBet-AF and SimBet-FR deliver about the same number of packets. However, under low traffic load, SimBet-AF is able to deliver more packets than SimBet-FR, because it can adapt to the changes in the network traffic load. Also, Figure 4.1d depicts the effect of the traffic load on the average route length of each algorithm.

As we can see in Figure 4.2a, SimBet-AF has about the same total number of packet drops as SimBet-FR. It should be noted that under high traffic load, SimBet-LB has more packet drops than the other two congestion control approaches, because the nodes



**Figure 4.2:** (a) Total Number of Packet Drops, (b) Cumulative Fraction of Packet Drops, (c) Overhead, and (d) Cumulative Fraction of Forwards on the Reality Mining data set

are unable to find alternative routes that are not under congestion. In Figure 4.2b we can clearly see that in the case of plain SimBet, a small subset of nodes is responsible for almost all the packet drops that occurred in the whole network. In particular, more than 80% of the total number of packet drops occurred in only 5 nodes, from a total of 97 nodes. Even the SimBet-FR algorithm exhibits this phenomenon to some degree, because it does not take into account the remaining storage space of the receiver before forwarding a packet. In contrast, with the SimBet-LB approach most of the nodes drop about the same amount of packets, but they are also more in total, which corresponds to fewer successfully delivered packets. The desired solution is given by the SimBet-AF algorithm, which combines the small number of packet drops with an almost fair distribution of the packet drops among the nodes.



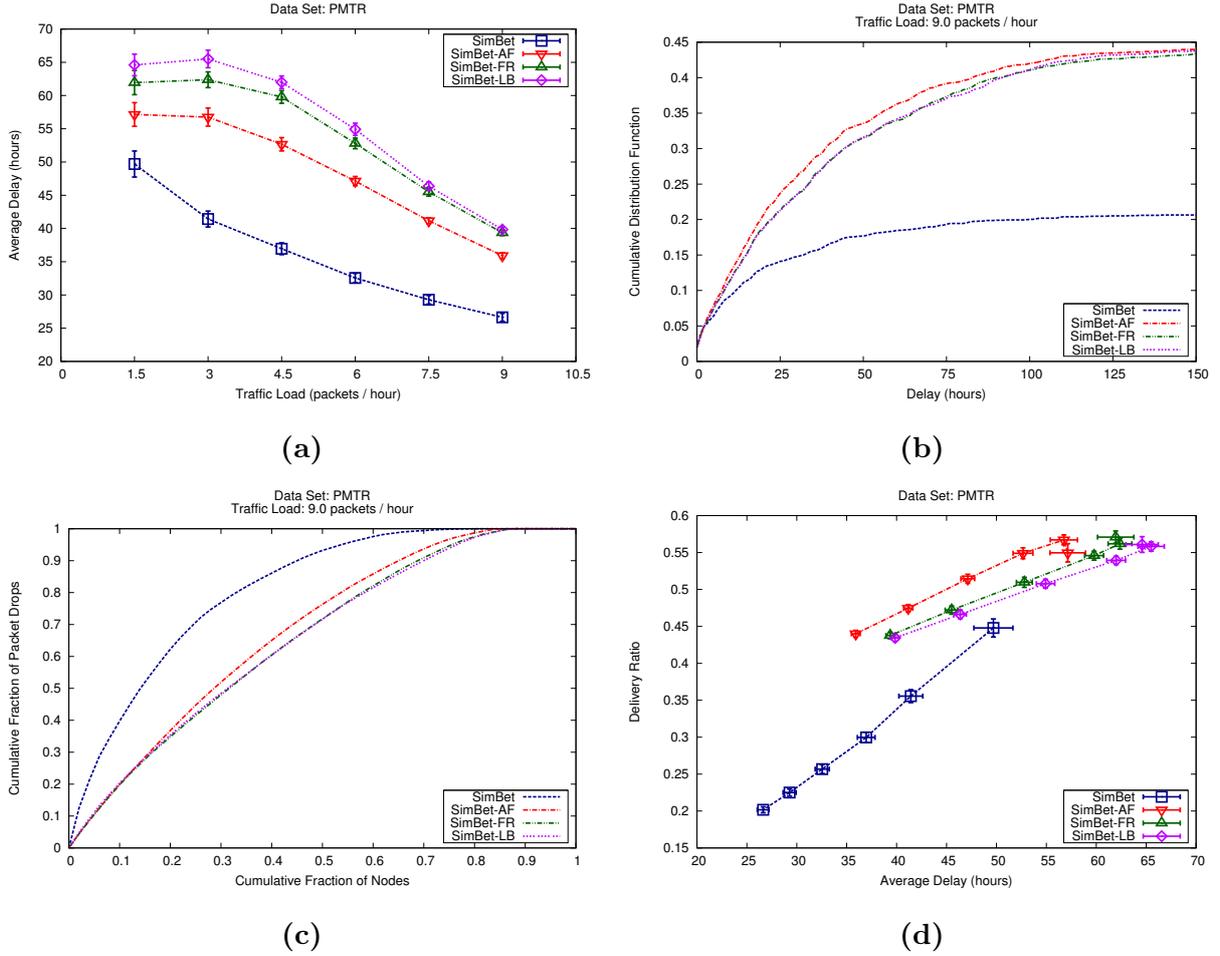
**Figure 4.3:** Average Delay vs. Delivery Ratio on the Reality Mining data set

Although the main goal of each algorithm is to deliver most of the packets in the minimum possible time, we also want this process to require a small amount of forwards. As we can see in Figure 4.2c, the SimBet-LB approach performs the fewest transmissions in the network, because of its notably stringent forwarding condition. It should be pointed out that plain SimBet delivered far fewer packets, while performing many more transmissions. Both SimBet-AF and SimBet-FR perform about the same number of forwards, which according to Figure 4.2d are distributed between the nodes in a similar way.

In conclusion, Figure 4.3 makes it clear that our proposed congestion control mechanism is the most efficient in terms of delivery ratio and delay. The queue control that was proposed by the authors of FairRoute, increases the delivery ratio to a certain degree, but the delay is significantly high compared to our proposed method. The approach that distributed the traffic load with absolute fairness caused the least overhead in the network, but it was not as efficient as the other approaches. We also showed how poorly SimBet performs under storage constraints without a proper congestion control mechanism, even under low traffic load.

### 4.3.2 Simulations on the PMTR and Sassy Data Sets

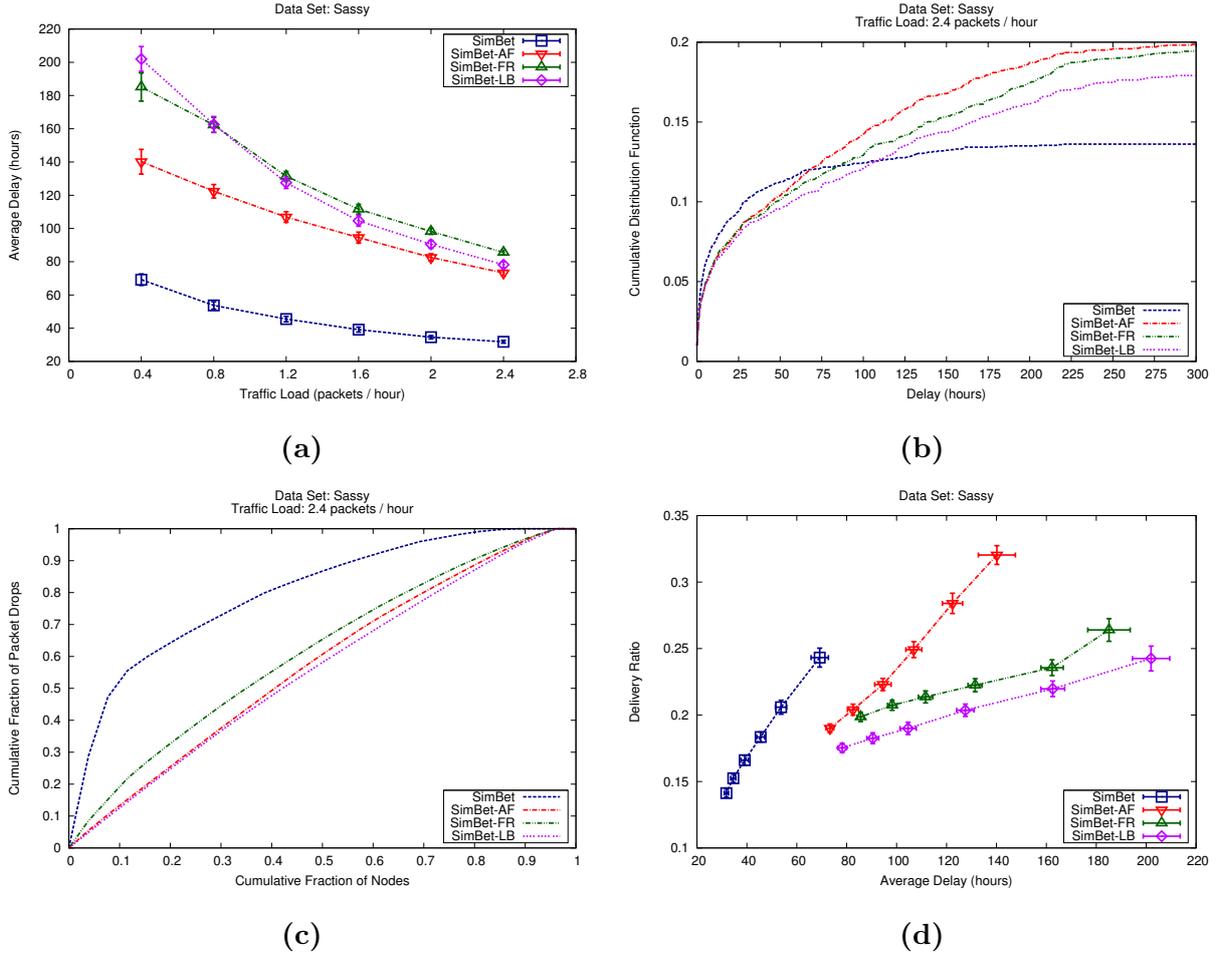
Since the characteristics of the other data sets are markedly different, we have to change the traffic load of each scenario in order to examine the performance of each algorithm



**Figure 4.4:** (a) Average Delay, (b) Cumulative Distribution Function of the Delay, (c) Cumulative Fraction of Packet Drops, and (d) Average Delay vs. Delivery Ratio on the PMTR data set

properly. However, the rest of the simulation settings remain the same, as well as the  $\delta$  value of the SimBet-AF algorithm, which is 0.25. The results that we obtained, both from the PMTR and Sassy data sets, were similar to those that we analyzed previously from the Reality Mining data set. Therefore, for these data sets we are going to analyze only the most important evaluation metrics.

As expected, plain SimBet has the lowest average delay in both data sets, as shown in Figures 4.4a and 4.5a, because it drops most of the packets due to congestion. Since there is only one copy of each packet in the network, plain SimBet could only deliver a small fraction of the generated packets, as we can see in Figures 4.4b and 4.5b. On the other hand, by including a congestion control mechanism, the total number of success-



**Figure 4.5:** (a) Average Delay, (b) Cumulative Distribution Function of the Delay, (c) Cumulative Fraction of Packet Drops, and (d) Average Delay vs. Delivery Ratio on the Sassy data set

fully delivered packets is significantly increased. Among the compared congestion control algorithms, SimBet-AF has the lowest average delay under any traffic load, while both SimBet-FR and SimBet-LB achieve similar average delivery times on both data sets.

As we can see in Figures 4.4c and 4.5c, plain SimBet causes most of its packet drops to occur in a small subset of nodes. This feature appeared in all three data sets, however in the Reality Mining data set we had the most skewed distribution. In the Sassy data set, SimBet-AF achieves almost as good distribution of the packet drops as SimBet-LB, while the SimBet-FR algorithm results in a slightly more skewed distribution. However, in the PMTR data set, all the algorithms have a similar distribution, with the exception of the plain SimBet.

Finally, according to Figures 4.3, 4.4d, and 4.5d, SimBet-AF was the most efficient in terms of delivery ratio and delay, in all three experimental data sets, under any traffic load. On the other hand, SimBet-FR also had a similar delivery ratio with SimBet-AF in all three data sets, but with much longer delivery times. The key component that makes our approach more efficient than the others, is the fact that we handle each forwarding decision differently. In order to decide if a packet should be forwarded, we take into account the saturation state of both nodes, as well as how much better the encountered node is than the current carrier, based on the utility values of the underlying routing algorithm. In addition to that, we can also adjust the fairness in the network, by tuning the value of the  $\delta$  parameter, in order to achieve the desired trade-off between efficiency and overhead.

# Chapter 5

## Conclusions and Future Work

### 5.1 Conclusions

In order to achieve data delivery in an Opportunistic Network, some routing protocols use utility functions to estimate the capability of each node to deliver a message to its destination. However, there is usually a small subset of important nodes that are more crucial than others in the process of data delivery. Congestion may occur in these nodes, even under low traffic load, if the messages are forwarded greedily. It is evident that a congestion control mechanism is needed that can take advantage of the most important nodes without excessively using them.

We proposed a congestion control mechanism that handles each message differently, in accordance with the saturation state and the utility values of the intermediate nodes. Additionally, a tunable parameter  $\delta$  was introduced in order to be able to adjust the fairness in the network. We can perceive the  $\delta$  parameter as the social preferences that each node has about allocating its resources to help others communicate. Altruistic nodes would have a  $\delta$  value close to 0, while selfish nodes would have the highest  $\delta$  values. However, in order to achieve the desired trade-off between efficiency and overhead, the  $\delta$  value should lie in the interval of  $(0, 1)$ , which corresponds to a cooperative behavior of the nodes.

Our experiments on three different real-world data sets showed that our congestion control mechanism reduces the average delay significantly, compared to other relevant approaches, while retaining a high ratio of delivered packets. In addition to that, our

approach does not overuse the most important nodes in the network, resulting in almost uniform distributions of packet drops and transmissions. As we concluded, we can significantly improve the performance of the network by slightly distorting its fairness in terms of resource allocation.

## 5.2 Future Work

During this work, the  $\delta$  parameter was the same fixed number for all the nodes in the network. We could extend the present approach, by setting the value of the  $\delta$  parameter dynamically, based on local information. For example, each node could define its own  $\delta$  value, based on its importance in the network or its preference to contribute as a router for others.

Social selfishness has also been considered as a potential problem in Opportunistic Networks [109, 110], where most of the nodes are willing to exchange messages only with those whom they have social relationships. In that case, each node could assign a different  $\delta$  value for every other node in the network, based on their social ties. Furthermore, the nodes could use buffer and network statistics in order to define the  $\delta$  value accordingly. For instance, each node could calculate the average time that packets destined for a certain node stay in its buffer, which may help to make even better use of the most important nodes.

The forwarding decisions that each node has to make could be considered as social dilemmas [111], where the nodes have to decide if they will act in favor of their own resources, or if they will contribute for the better performance of the network. Therefore, we could apply game theory [112] in order to make the required forwarding decisions that aim to solve the congestion control problem.

# Bibliography

- [1] N. Abramson, “THE ALOHA SYSTEM: Another alternative for computer communications,” in *Proceedings of the 1970 Fall Joint Computer Conference (AFIPS '70)*, 1970, pp. 281–285.
- [2] —, “Development of the ALOHANET,” *IEEE Transactions on Information Theory*, vol. 31, no. 2, pp. 119–123, 1985.
- [3] B. P. Crow, I. Widjaja, J. G. Kim, and P. T. Sakai, “IEEE 802.11 wireless local area networks,” *IEEE Communications Magazine*, vol. 35, no. 9, pp. 116–126, 1997.
- [4] G. R. Hiertz, D. Denteneer, L. Stibor, Y. Zang, X. P. Costa, and B. Walke, “The IEEE 802.11 universe,” *IEEE Communications Magazine*, vol. 48, no. 1, pp. 62–70, 2010.
- [5] R. E. Kahn, “The organization of computer resources into a packet radio network,” *IEEE Transactions on Communications*, vol. 25, no. 1, pp. 169–178, 1977.
- [6] J. Jubin and J. D. Tornow, “The DARPA packet radio network protocols,” *Proceedings of the IEEE*, vol. 75, no. 1, pp. 21–32, 1987.
- [7] C. E. Perkins and P. Bhagwat, “Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers,” *SIGCOMM Computer Communication Review*, vol. 24, no. 4, pp. 234–244, 1994.
- [8] D. B. Johnson and D. A. Maltz, “Dynamic source routing in ad hoc wireless networks,” in *Mobile Computing*, T. Imieliński and H. F. Korth, Eds. Kluwer Academic Publishers, 1996, ch. 5, pp. 153–181.

- [9] C. E. Perkins and E. M. Royer, “Ad-hoc on-demand distance vector routing,” in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99)*, 1999, pp. 90–100.
- [10] Y.-B. Ko and N. H. Vaidya, “Location-aided routing (LAR) in mobile ad hoc networks,” in *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '98)*, 1998, pp. 66–75.
- [11] M. Joa-Ng and I.-T. Lu, “A peer-to-peer zone-based two-level link state routing for mobile ad hoc networks,” *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1415–1425, 1999.
- [12] P. Jacquet, P. Mühlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, “Optimized link state routing protocol for ad hoc networks,” in *Proceedings of the 5th IEEE International Multi Topic Conference (INMIC '01)*, 2001, pp. 62–68.
- [13] E. M. Royer and C.-K. Toh, “A review of current routing protocols for ad hoc mobile wireless networks,” *IEEE Personal Communications*, vol. 6, no. 2, pp. 46–55, 1999.
- [14] M. Abolhasan, T. Wysocki, and E. Dutkiewicz, “A review of routing protocols for mobile ad hoc networks,” *Ad Hoc Networks*, vol. 2, no. 1, pp. 1–22, 2004.
- [15] J. Hoebeke, I. Moerman, B. Dhoedt, and P. Demeester, “An overview of mobile ad hoc networks: Applications and challenges,” *Journal of the Communications Network*, vol. 3, no. 3, pp. 60–66, 2004.
- [16] Z. Zhang, “Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: Overview and challenges,” *IEEE Communications Surveys & Tutorials*, vol. 8, no. 1, pp. 24–37, 2006.
- [17] K. Fall, “A delay-tolerant network architecture for challenged internets,” in *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '03)*, 2003, pp. 27–34.
- [18] A. Pentland, R. Fletcher, and A. Hasson, “DakNet: Rethinking connectivity in developing nations,” *Computer*, vol. 37, no. 1, pp. 78–83, 2004.

- [19] E. Brewer, M. Demmer, B. Du, M. Ho, M. Kam, S. Nedeveschi, J. Pal, R. Patra, S. Surana, and K. Fall, “The case for technology in developing regions,” *Computer*, vol. 38, no. 6, pp. 25–38, 2005.
- [20] A. Doria, M. Uden, and D. P. Pandey, “Providing connectivity to the saami nomadic community,” in *Proceedings of the 2nd International Conference on Open Collaborative Design for Sustainable Innovation (dyd '02)*, 2002.
- [21] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, “MaxProp: Routing for vehicle-based disruption-tolerant networks,” in *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM '06)*, 2006, pp. 1–11.
- [22] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden, “CarTel: A distributed mobile sensor computing system,” in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys '06)*, 2006, pp. 125–138.
- [23] J. Ott and D. Kutscher, “Drive-thru internet: IEEE 802.11b for "automobile" users,” in *Proceedings of the 23rd IEEE International Conference on Computer Communications (INFOCOM '04)*, 2004.
- [24] P. Juang, H. Oki, Y. Wang, M. Martonosi, L.-S. Peh, and D. Rubenstein, “Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with ZebraNet,” in *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS X)*, 2002, pp. 96–107.
- [25] T. Small and Z. J. Haas, “The shared wireless infostation model: A new ad hoc networking paradigm (or where there is a whale, there is a way),” in *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '03)*, 2003, pp. 233–244.
- [26] R. C. Shah, S. Roy, S. Jain, and W. Brunette, “Data MULEs: modeling and analysis of a three-tier architecture for sparse sensor networks,” *Ad Hoc Networks*, vol. 1, no. 2–3, pp. 215–233, 2003.

- [27] R. Krishnan, P. Basu, J. M. Mikkelsen, C. Small, R. Ramanathan, D. W. Brown, J. R. Burgess, A. L. Caro, M. Condell, N. C. Goffee, R. R. Hain, R. E. Hansen, C. E. Jones, V. Kawadia, D. P. Mankins, B. I. Schwartz, W. T. Strayer, J. W. Ward, D. P. Wiggins, and S. H. Polit, “The SPINDLE disruption-tolerant networking system,” in *Proceedings of the 2007 IEEE Military Communications Conference (MILCOM '07)*, 2007, pp. 1–7.
- [28] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, and H. Weiss, “Delay-tolerant networking: An approach to interplanetary internet,” *IEEE Communications Magazine*, vol. 41, no. 6, pp. 128–136, 2003.
- [29] M. Grossglauser and D. Tse, “Mobility increases the capacity of ad-hoc wireless networks,” in *Proceedings of the 12th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '01)*, 2001, pp. 1360–1369.
- [30] S. Jain, K. Fall, and R. Patra, “Routing in a delay tolerant network,” in *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '04)*, 2004, pp. 145–158.
- [31] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot, “Pocket switched networks and human mobility in conference environments,” in *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN '05)*, 2005, pp. 244–251.
- [32] T. Spyropoulos, T. Turletti, and K. Obraczka, “Routing in delay-tolerant networks comprising heterogeneous node populations,” *IEEE Transactions on Mobile Computing*, vol. 8, no. 8, pp. 1132–1147, 2009.
- [33] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Efficient routing in intermittently connected mobile networks: The single-copy case,” *IEEE/ACM Transactions on Networking*, vol. 16, no. 1, pp. 63–76, 2008.
- [34] —, “Efficient routing in intermittently connected mobile networks: The multiple-copy case,” *IEEE/ACM Transactions on Networking*, vol. 16, no. 1, pp. 77–90, 2008.

- [35] M. Musolesi and C. Mascolo, “CAR: Context-aware adaptive routing for delay-tolerant mobile networks,” *IEEE Transactions on Mobile Computing*, vol. 8, no. 2, pp. 246–260, 2009.
- [36] J. Whitbeck and V. Conan, “HYMAD: Hybrid DTN-MANET routing for dense and highly dynamic wireless networks,” *Computer Communications*, vol. 33, no. 13, pp. 1483–1492, 2010.
- [37] L. Pelusi, A. Passarella, and M. Conti, “Opportunistic networking: Data forwarding in disconnected mobile ad hoc networks,” *IEEE Communications Magazine*, vol. 44, no. 11, pp. 134–141, 2006.
- [38] T. Spyropoulos, R. N. B. Rais, T. Turetli, K. Obraczka, and A. Vasilakos, “Routing for disruption tolerant networks: taxonomy and design,” *Wireless Networks*, vol. 16, no. 8, pp. 2349–2370, 2010.
- [39] A. Vahdat and D. Becker, “Epidemic routing for partially connected ad hoc networks,” Duke University, Tech. Rep. CS-200006, 2000.
- [40] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, “The broadcast storm problem in a mobile ad hoc network,” in *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom ’99)*, 1999, pp. 151–162.
- [41] X. Zhang, G. Neglia, J. Kurose, and D. Towsley, “Performance modeling of epidemic routing,” *Computer Networks*, vol. 51, no. 10, pp. 2867–2891, 2007.
- [42] S. C. Nelson, M. Bakht, and R. Kravets, “Encounter-based routing in DTNs,” in *Proceedings of the 28th IEEE International Conference on Computer Communications (INFOCOM ’09)*, 2009, pp. 846–854.
- [43] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Spray and wait: An efficient routing scheme for intermittently connected mobile networks,” in *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN ’05)*, 2005, pp. 252–259.

- [44] —, “Spray and focus: Efficient mobility-assisted routing for heterogeneous and correlated mobility,” in *Proceedings of the 5th Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom '07)*, 2007, pp. 79–85.
- [45] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, “Network information flow,” *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, 2000.
- [46] C. Fragouli, J.-Y. Le Boudec, and J. Widmer, “Network coding: An instant primer,” *SIGCOMM Computer Communication Review*, vol. 36, no. 1, pp. 63–68, 2006.
- [47] J. Widmer and J.-Y. Le Boudec, “Network coding for efficient communication in extreme networks,” in *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN '05)*, 2005, pp. 284–291.
- [48] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, “XORs in the air: Practical wireless network coding,” in *Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '06)*, 2006, pp. 243–254.
- [49] A. Lindgren, A. Doria, and O. Schelén, “Probabilistic routing in intermittently connected networks,” *SIGMOBILE Mobile Computing and Communications Review*, vol. 7, no. 3, pp. 19–20, 2003.
- [50] —, “Probabilistic routing in intermittently connected networks,” in *Service Assurance with Partial and Intermittent Resources*, ser. Lecture Notes in Computer Science, P. Dini, P. Lorenz, and J. N. de Souza, Eds. Springer Berlin Heidelberg, 2004, vol. 3126, pp. 239–254.
- [51] S. Grasic, E. Davies, A. Lindgren, and A. Doria, “The evolution of a DTN routing protocol - PRoPHETv2,” in *Proceedings of the 6th ACM Workshop on Challenged Networks (CHANTS '11)*, 2011, pp. 27–30.
- [52] E. C. R. de Oliveira and C. V. N. de Albuquerque, “NECTAR: A DTN routing protocol based on neighborhood contact history,” in *Proceedings of the 24th ACM Symposium on Applied Computing (SAC '09)*, 2009, pp. 40–46.

- [53] E. P. C. Jones, L. Li, and P. A. S. Ward, “Practical routing in delay-tolerant networks,” in *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN '05)*, 2005, pp. 237–243.
- [54] M. C. González, C. A. Hidalgo, and A.-L. Barabási, “Understanding individual human mobility patterns,” *Nature*, vol. 453, no. 7196, pp. 779–782, 2008.
- [55] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási, “Limits of predictability in human mobility,” *Science*, vol. 327, no. 5968, pp. 1018–1021, 2010.
- [56] 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 *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM '06)*, 2006, pp. 1–13.
- [57] T. Karagiannis, J.-Y. Le Boudec, and M. Vojnović, “Power law and exponential decay of inter contact times between mobile devices,” in *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking (MobiCom '07)*, 2007, pp. 183–194.
- [58] T. Hossmann, T. Spyropoulos, and F. Legendre, “A complex network analysis of human mobility,” in *Proceedings of the 2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2011, pp. 876–881.
- [59] S. Milgram, “The small-world problem,” *Psychology today*, vol. 1, no. 1, pp. 61–67, 1967.
- [60] I. de Sola Pool and M. Kochen, “Contacts and influence,” *Social Networks*, vol. 1, no. 1, pp. 5–51, 1978–1979.
- [61] W. Hsu and A. Helmy, “On nodal encounter patterns in wireless LAN traces,” in *Proceedings of the 4th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt '06)*, 2006, pp. 1–10.
- [62] D. J. Watts and S. H. Strogatz, “Collective dynamics of ‘small-world’ networks,” *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

- [63] D. J. Watts, “Networks, dynamics, and the small-world phenomenon,” *American Journal of Sociology*, vol. 105, no. 2, pp. 493–527, 1999.
- [64] A. Barrat and M. Weigt, “On the properties of small-world network models,” *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 13, no. 3, pp. 547–560, 2000.
- [65] L. C. Freeman, “A set of measures of centrality based on betweenness,” *Sociometry*, vol. 40, no. 1, pp. 35–41, 1977.
- [66] —, “Centrality in social networks conceptual clarification,” *Social Networks*, vol. 1, no. 3, pp. 215–239, 1978–1979.
- [67] D. Liben-Nowell and J. Kleinberg, “The link prediction problem for social networks,” in *Proceedings of the 12th International Conference on Information and Knowledge Management (CIKM '03)*, 2003, pp. 556–559.
- [68] M. E. J. Newman, “A measure of betweenness centrality based on random walks,” *Social Networks*, vol. 27, no. 1, pp. 39–54, 2005.
- [69] E. M. Daly and M. Haahr, “Social network analysis for routing in disconnected delay-tolerant MANETs,” in *Proceedings of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '07)*, 2007, pp. 32–40.
- [70] P. V. Marsden, “Egocentric and sociocentric measures of network centrality,” *Social Networks*, vol. 24, no. 4, pp. 407–422, 2002.
- [71] E. M. Daly and M. Haahr, “Social network analysis for information flow in disconnected delay-tolerant MANETs,” *IEEE Transactions on Mobile Computing*, vol. 8, no. 5, pp. 606–621, 2009.
- [72] P. Hui, J. Crowcroft, and E. Yoneki, “Bubble rap: Social-based forwarding in delay tolerant networks,” in *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '08)*, 2008, pp. 241–250.
- [73] —, “Bubble rap: Social-based forwarding in delay-tolerant networks,” *IEEE Transactions on Mobile Computing*, vol. 10, no. 11, pp. 1576–1589, 2011.

- [74] M. E. J. Newman, “Analysis of weighted networks,” *Physical Review E*, vol. 70, no. 5, p. 056131, 2004.
- [75] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, “Uncovering the overlapping community structure of complex networks in nature and society,” *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.
- [76] P. Hui, E. Yoneki, S.-Y. Chan, and J. Crowcroft, “Distributed community detection in delay tolerant networks,” in *Proceedings of the 2nd ACM/IEEE International Workshop on Mobility in the Evolving Internet Architecture (MobiArch '07)*, 2007, pp. 7:1–7:8.
- [77] E. Bulut and B. K. Szymanski, “Friendship based routing in delay tolerant mobile social networks,” in *Proceedings of the 2010 IEEE Global Telecommunications Conference (GLOBECOM '10)*, 2010, pp. 1–5.
- [78] —, “Exploiting friendship relations for efficient routing in mobile social networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 12, pp. 2254–2265, 2012.
- [79] A. Lindgren and K. S. Phanse, “Evaluation of queueing policies and forwarding strategies for routing in intermittently connected networks,” in *Proceedings of the 1st International Conference on Communication System Software and Middleware (COMSWARE '06)*, 2006, pp. 1–10.
- [80] V. Erramilli and M. Crovella, “Forwarding in opportunistic networks with resource constraints,” in *Proceedings of the 3rd ACM Workshop on Challenged Networks (CHANTS '08)*, 2008, pp. 41–48.
- [81] A. Krifa, C. Barakat, and T. Spyropoulos, “Optimal buffer management policies for delay tolerant networks,” in *Proceedings of the 5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '08)*, 2008, pp. 260–268.
- [82] —, “Message drop and scheduling in DTNs: Theory and practice,” *IEEE Transactions on Mobile Computing*, vol. 11, no. 9, pp. 1470–1483, 2012.

- [83] A. Balasubramanian, B. N. Levine, and A. Venkataramani, “DTN routing as a resource allocation problem,” *SIGCOMM Computer Communication Review*, vol. 37, no. 4, pp. 373–384, 2007.
- [84] ———, “Replication routing in DTNs: A resource allocation approach,” *IEEE/ACM Transactions on Networking*, vol. 18, no. 2, pp. 596–609, 2010.
- [85] G. Sandulescu and S. Nadjm-Tehrani, “Opportunistic DTN routing with window-aware adaptive replication,” in *Proceedings of the 4th Asian Conference on Internet Engineering (AINTEC '08)*, 2008, pp. 103–112.
- [86] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot, “Delegation forwarding,” in *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '08)*, 2008, pp. 251–260.
- [87] B. Soelistijanto and M. P. Howarth, “Transfer reliability and congestion control strategies in opportunistic networks: A survey,” *IEEE Communications Surveys & Tutorials*, vol. PP, no. 99, pp. 1–18, 2013.
- [88] V. Erramilli, A. Chaintreau, M. Crovella, and C. Diot, “Diversity of forwarding paths in pocket switched networks,” in *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement (IMC '07)*, 2007, pp. 161–174.
- [89] J. M. Pujol, A. L. Toledo, and P. Rodriguez, “Fair routing in delay tolerant networks,” in *Proceedings of the 28th IEEE International Conference on Computer Communications (INFOCOM '09)*, 2009, pp. 837–845.
- [90] K. Fall, W. Hong, and S. Madden, “Custody transfer for reliable delivery in delay tolerant networks,” Intel Research Berkeley, Tech. Rep. IRB-TR-03-030, 2003.
- [91] S. Burleigh, E. Jennings, and J. Schoolcraft, “Autonomous congestion control in delay-tolerant networks,” in *Proceedings of the 9th AIAA International Conference on Space Operations (SpaceOps '06)*, 2006.
- [92] M. Radenkovic and A. Grundy, “Congestion aware forwarding in delay tolerant and social opportunistic networks,” in *Proceedings of the 8th International Conference*

- on *Wireless On-Demand Network Systems and Services (WONS '11)*, 2011, pp. 60–67.
- [93] M. Seligman, K. Fall, and P. Mundur, “Alternative custodians for congestion control in delay tolerant networks,” in *Proceedings of the 2006 SIGCOMM Workshop on Challenged Networks (CHANTS '06)*, 2006, pp. 229–236.
- [94] —, “Storage routing for DTN congestion control,” *Wireless Communications and Mobile Computing*, vol. 7, no. 10, pp. 1183–1196, 2007.
- [95] T. Small and Z. J. Haas, “Resource and performance tradeoffs in delay-tolerant wireless networks,” in *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN '05)*, 2005, pp. 260–267.
- [96] N. Thompson, S. C. Nelson, M. Bakht, T. Abdelzaher, and R. Kravets, “Retiring replicants: Congestion control for intermittently-connected networks,” in *Proceedings of the 29th IEEE International Conference on Computer Communications (INFOCOM '10)*, 2010, pp. 1–9.
- [97] M. Radenkovic and A. Grundy, “Framework for utility driven congestion control in delay tolerant opportunistic networks,” in *Proceedings of the 7th International Wireless Communications and Mobile Computing Conference (IWCMC '11)*, 2011, pp. 448–454.
- [98] N. Eagle and A. Pentland, “Reality mining: sensing complex social systems,” *Personal and Ubiquitous Computing*, vol. 10, no. 4, pp. 255–268, 2006.
- [99] A. J. Mashhadi, S. B. Mokhtar, and L. Capra, “Fair content dissemination in participatory DTNs,” *Ad Hoc Networks*, vol. 10, no. 8, pp. 1633–1645, 2012.
- [100] B. Soelistijanto and M. Howarth, “Traffic distribution and network capacity analysis in social opportunistic networks,” in *Proceedings of the 8th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob '12)*, 2012, pp. 823–830.
- [101] A. Mtibaa and K. A. Harras, “Fairness-related challenges in mobile opportunistic networking,” *Computer Networks*, vol. 57, no. 1, pp. 228–242, 2013.

- [102] D. M. Messick and C. G. McClintock, “Motivational bases of choice in experimental games,” *Journal of Experimental Social Psychology*, vol. 4, no. 1, pp. 1–25, 1968.
- [103] D. W. Griesinger and J. W. Livingston, “Toward a model of interpersonal motivation in experimental games,” *Behavioral Science*, vol. 18, no. 3, pp. 173–188, 1973.
- [104] W. B. G. Liebrand, “The effect of social motives, communication and group size on behaviour in an n-person multi-stage mixed-motive game,” *European Journal of Social Psychology*, vol. 14, no. 3, pp. 239–264, 1984.
- [105] P. A. M. Van Lange, E. M. N. De Bruin, W. Otten, and J. A. Joireman, “Development of prosocial, individualistic, and competitive orientations: Theory and preliminary evidence,” *Journal of Personality and Social Psychology*, vol. 73, no. 4, pp. 733–746, 1997.
- [106] “Crawdad: A community resource for archiving wireless data at dartmouth.” [Online]. Available: <http://crawdad.cs.dartmouth.edu/>
- [107] S. Gaito, E. Pagani, and G. P. Rossi, “Opportunistic forwarding in workplaces,” in *Proceedings of the 2nd ACM Workshop on Online Social Networks (WOSN '09)*, 2009, pp. 55–60.
- [108] G. Bigwood, D. Rehunathan, M. Bateman, T. Henderson, and S. Bhatti, “Exploiting self-reported social networks for routing in ubiquitous computing environments,” in *Proceedings of the 4th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob '08)*, 2008, pp. 484–489.
- [109] Q. Li, S. Zhu, and G. Cao, “Routing in socially selfish delay tolerant networks,” in *Proceedings of the 29th IEEE International Conference on Computer Communications (INFOCOM '10)*, 2010, pp. 1–9.
- [110] Q. Li, W. Gao, S. Zhu, and G. Cao, “A routing protocol for socially selfish delay tolerant networks,” *Ad Hoc Networks*, vol. 10, no. 8, pp. 1619–1632, 2012.
- [111] N. L. Kerr, “Motivation losses in small groups: A social dilemma analysis,” *Journal of Personality and Social Psychology*, vol. 45, no. 4, pp. 819–828, 1983.

- [112] R. B. Myerson, *Game Theory: Analysis of Conflict*. Harvard University Press, 1991.
- [113] T. Hossmann, T. Spyropoulos, and F. Legendre, “Know thy neighbor: Towards optimal mapping of contacts to social graphs for DTN routing,” in *Proceedings of the 29th IEEE International Conference on Computer Communications (INFOCOM '10)*, 2010, pp. 1–9.
- [114] M. Everett and S. P. Borgatti, “Ego network betweenness,” *Social Networks*, vol. 27, no. 1, pp. 31–38, 2005.

# Appendix A

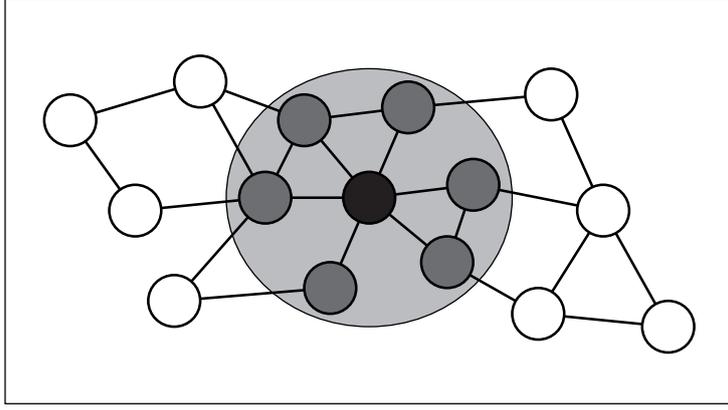
## The SimBet Algorithm

SimBet [69] is a social-based routing protocol for Opportunistic Networks that relies on two social metrics, betweenness centrality and similarity. In order to be able to calculate these metrics, a social graph is required that describes the social relations between the nodes of the network. There are several techniques of contact aggregation that aim to map the physical encounters of the nodes to social graphs [113]. The original version of SimBet uses a growing time window to build its social graphs, by adding edges between nodes that have met at least once in the past.

Centrality is a measure of the relative importance of a node within a graph. There are several centrality definitions, however the most common are degree centrality, closeness centrality, and betweenness centrality [65, 66]. SimBet exploits the betweenness centrality, because it can be regarded as a quantify of the control that a node has over information flowing between others [68]. The betweenness centrality of node  $p_i$  is equal to the number of shortest paths between any pair of nodes  $p_j$  and  $p_k$  that pass through node  $p_i$ , divided by the total number of shortest paths, as seen in Equation A.1.

$$C_B(p_i) = \sum_{j=1}^N \sum_{k=1}^{j-1} \frac{g_{jk}(p_i)}{g_{jk}} \quad (\text{A.1})$$

Unfortunately, the betweenness centrality measure requires complete knowledge of the network topology, which is not available in Opportunistic Networks due to their intermittent connectivity. Therefore, SimBet calculates the betweenness centrality of the ego network (egocentric betweenness), rather than the betweenness centrality of the complete network (sociocentric betweenness) [70]. An ego network consists of the ego



**Figure A.1:** The gray area of the given social graph corresponds to the ego network of the black node

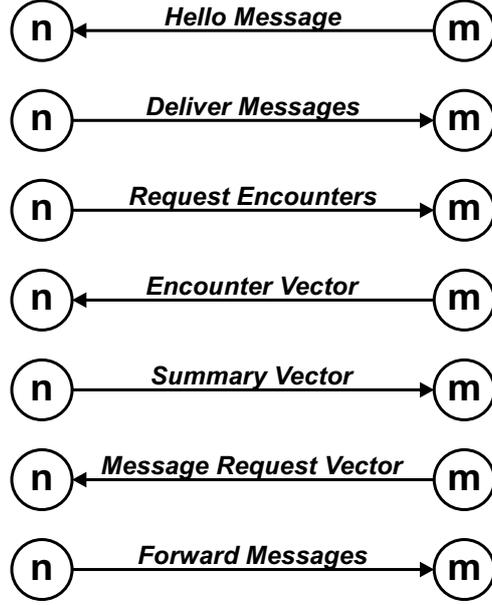
node, its neighbors, and all the links among them, as illustrated in Figure A.1. Given the adjacency matrix of the ego network  $A$ , the sum of the reciprocals of the entries of  $A^2[1 - A]$  is equal to the egocentric betweenness of the ego node [114]. Although the egocentric betweenness of a node is generally smaller than its sociocentric betweenness, it has been observed that the ranking of the nodes is similar for both measures.

Social networks are known for their high clustering coefficient and short average path lengths [62, 63, 64]. Based on this observation, a variety of metrics that rely on neighborhoods of nodes have been proposed, in order to predict future links [67]. SimBet uses the number of common neighbors, as shown in Equation A.2, to calculate the similarity metric between two nodes  $x$  and  $y$ .

$$P(x, y) = |N(x) \cap N(y)| \quad (\text{A.2})$$

Therefore, based on the adjacency matrix of the ego network, the ego node can calculate its similarity with any node that has met directly. However, in order to calculate the similarity of the ego node with nodes that do not belong in its ego network, we need a list of indirect encounters through its neighbors. When two nodes meet, they exchange their lists of direct encounters, from which each node can obtain information about its indirect encounters that can be used for the calculation of the similarity metric.

Each node periodically transmits “hello” messages, so that it can be detected by the other nodes. Whenever a node receives a “hello” message from another node, which indicates that they are within communication range, it delivers all the messages that it



**Figure A.2:** Exchange of Messages in SimBet

carries that are destined for that node. Then, it requests the encounter vector of the other node, so that it can update its ego network and social metrics. After the reception of the encounter vector, it transmits a summary vector that contains a summary of all the messages that it holds. Finally, the other node decides which of these messages should be transmitted, based on their utilities. This process is also described in Figure A.2.

In order to decide which messages should be forwarded, the two nodes have to compare their utilities. The similarity utility is a comparison of the similarity metrics that nodes  $n$  and  $m$  have for the destination node  $d$ . Likewise, the betweenness utility is a comparison of the egocentric betweenness of the nodes  $n$  and  $m$ . Equations A.3 and A.4 define how node  $n$  calculates its similarity utility and betweenness utility for delivering a message to node  $d$ , compared to node  $m$ . The SimBet utility is calculated as a weighted combination of the similarity utility and the betweenness utility, as given in Equation A.5.

$$SimUtil_n(d) = \frac{Sim_n(d)}{Sim_n(d) + Sim_m(d)} \quad (\text{A.3})$$

$$BetUtil_n = \frac{Bet_n}{Bet_n + Bet_m} \quad (\text{A.4})$$

$$SimBetUtil_n(d) = \alpha SimUtil_n(d) + \beta BetUtil_n \quad (\text{A.5})$$

The SimBet utility is used to make the forwarding decisions, where  $\alpha$  and  $\beta$  are adjustable parameters for which it holds that  $\alpha + \beta = 1$ . These parameters are usually set to  $\alpha = \beta = 0.5$ , so that the similarity utility and the betweenness utility have the same importance. Finally, node  $n$  will forward messages, destined for node  $d$ , to node  $m$  only if  $SimBetUtil_m(d) > SimBetUtil_n(d)$ .

# Appendix B

## The FairRoute Algorithm

FairRoute [89] is a routing protocol for Opportunistic Networks that uses the interaction strength between two nodes  $i$  and  $j$ , in short term  $\sigma_{ij}$  and in long term  $\lambda_{ij}$ , in order to make its routing decisions. The interaction strength increases upon encounter, while it decreases over time with an exponential rate  $r_\sigma$  for short term and  $r_\lambda$  for long term, which means that these parameters must be set so that  $r_\lambda \ll r_\sigma$ . Equations B.1, B.2, and B.3 define how node  $i$  updates its perceived interaction strengths when it encounters node  $j$ , where  $t$  is the current time,  $t_i$  is the last time that node  $i$  encountered another node, and  $N_i$  is the list of contacts of node  $i$ .

$$\sigma_{ik} = \sigma_{ik} e^{-r_\sigma(t-t_i)} \quad \forall k \in N_i \quad (\text{B.1})$$

$$\lambda_{ik} = \lambda_{ik} e^{-r_\lambda(t-t_i)} \quad \forall k \in N_i \quad (\text{B.2})$$

$$(\sigma_{ij}, \lambda_{ij}) = (\sigma_{ij}, \lambda_{ij}) + (1, 1) \quad (\text{B.3})$$

After updating the perceived interaction strengths, the two nodes have to decide which messages should be forwarded from the one node to the other. FairRoute favors nodes with high interaction strength in long term, while avoiding nodes with deceptive interaction strength in short term. Based on their perceived interaction strengths, node  $i$  can calculate its perceived utility of node  $j$  to deliver a message to node  $k$ , as defined in Equation B.4. In a similar way, node  $i$  calculates its perceived utility of node  $j$  to deliver a message to any node in the network, as shown in Equation B.5. Consequently, node  $i$  will forward a

message to node  $j$ , which is destined for node  $k$ , only if at least one of the conditions of Equation B.6 is met.

$$u_{ijk} = \frac{\lambda_{jk}(\lambda_{jk} - \sigma_{jk})}{\lambda_{jk}(\lambda_{jk} - \sigma_{jk}) + \lambda_{ik}(\lambda_{ik} - \sigma_{ik})} \quad (\text{B.4})$$

$$u_{ij} = \frac{\sum_{k \in N_j} \lambda_{jk}(\lambda_{jk} - \sigma_{jk})}{\sum_{k \in N_j} \lambda_{jk}(\lambda_{jk} - \sigma_{jk}) + \sum_{k \in N_i} \lambda_{ik}(\lambda_{ik} - \sigma_{ik})} \quad (\text{B.5})$$

$$\begin{cases} u_{ijk} > \frac{1}{2} & \wedge & (\lambda_{ik} + \lambda_{jk}) > 0 \\ u_{ij} > \frac{1}{2} & \wedge & (\lambda_{ik} + \lambda_{jk}) = 0 \end{cases} \quad (\text{B.6})$$

However, this routing strategy is forwarding each message greedily, like most other routing protocols, causing an unfair traffic load distribution. To overcome this problem, FairRoute uses a queue control that takes into account the queue length of each node. According to this queue control, each node will accept messages only from nodes with higher or equal queue lengths, except from the case where  $u_{ijk} = 1$  and  $(\lambda_{ik} + \lambda_{jk}) > 0$ . Finally, Equation B.7 modifies the conditions of Equation B.6, by including the queue control.

$$\begin{cases} u_{ijk} > \frac{1}{2} & \wedge & (\lambda_{ik} + \lambda_{jk}) > 0 & \wedge & Q_j \leq Q_i \\ u_{ijk} = 1 & \wedge & (\lambda_{ik} + \lambda_{jk}) > 0 \\ u_{ij} > \frac{1}{2} & \wedge & (\lambda_{ik} + \lambda_{jk}) = 0 & \wedge & Q_j \leq Q_i \end{cases} \quad (\text{B.7})$$

This queue control can also be used in other utility-based routing protocols to increase their fairness, since it is independent from the routing decisions. The authors of FairRoute argue that the queue size of a node is equivalent to its social status and therefore the nodes should not accept messages from nodes of lower social status.

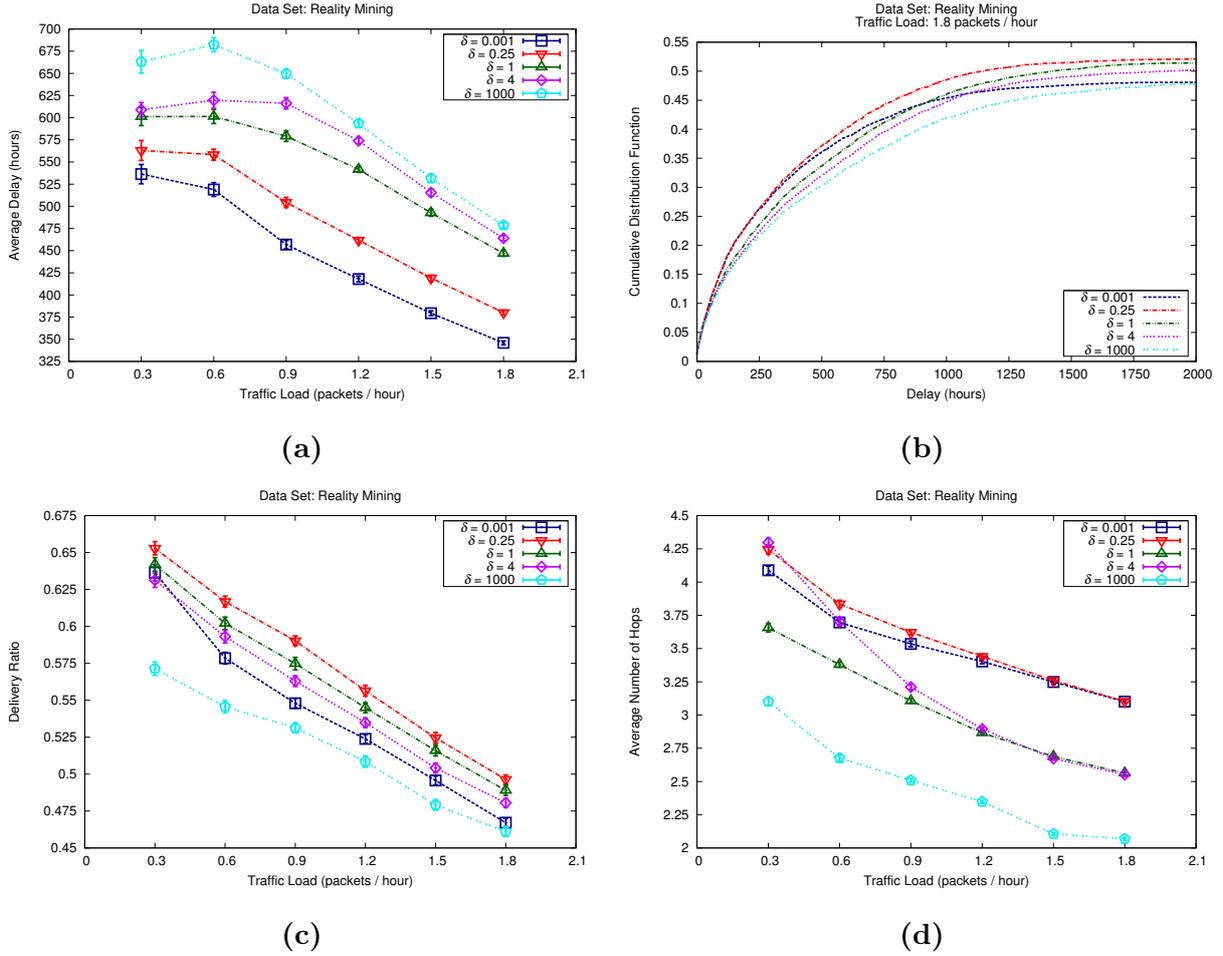
# Appendix C

## The Impact of Social Preferences

The  $\delta$  parameter of our proposed congestion control mechanism can be used to define the social preferences of the nodes. When  $\delta$  is close to 0, the nodes behave altruistically by accepting to carry most of the forwarding packets. Alternatively, when a high value is assigned to the  $\delta$  parameter, the nodes behave selfishly by rejecting to carry most of the forwarding packets. In order to examine the impact of five different  $\delta$  values on the performance of the network, we are going to evaluate our congestion control mechanism on the single-copy version of the SimBet routing protocol.

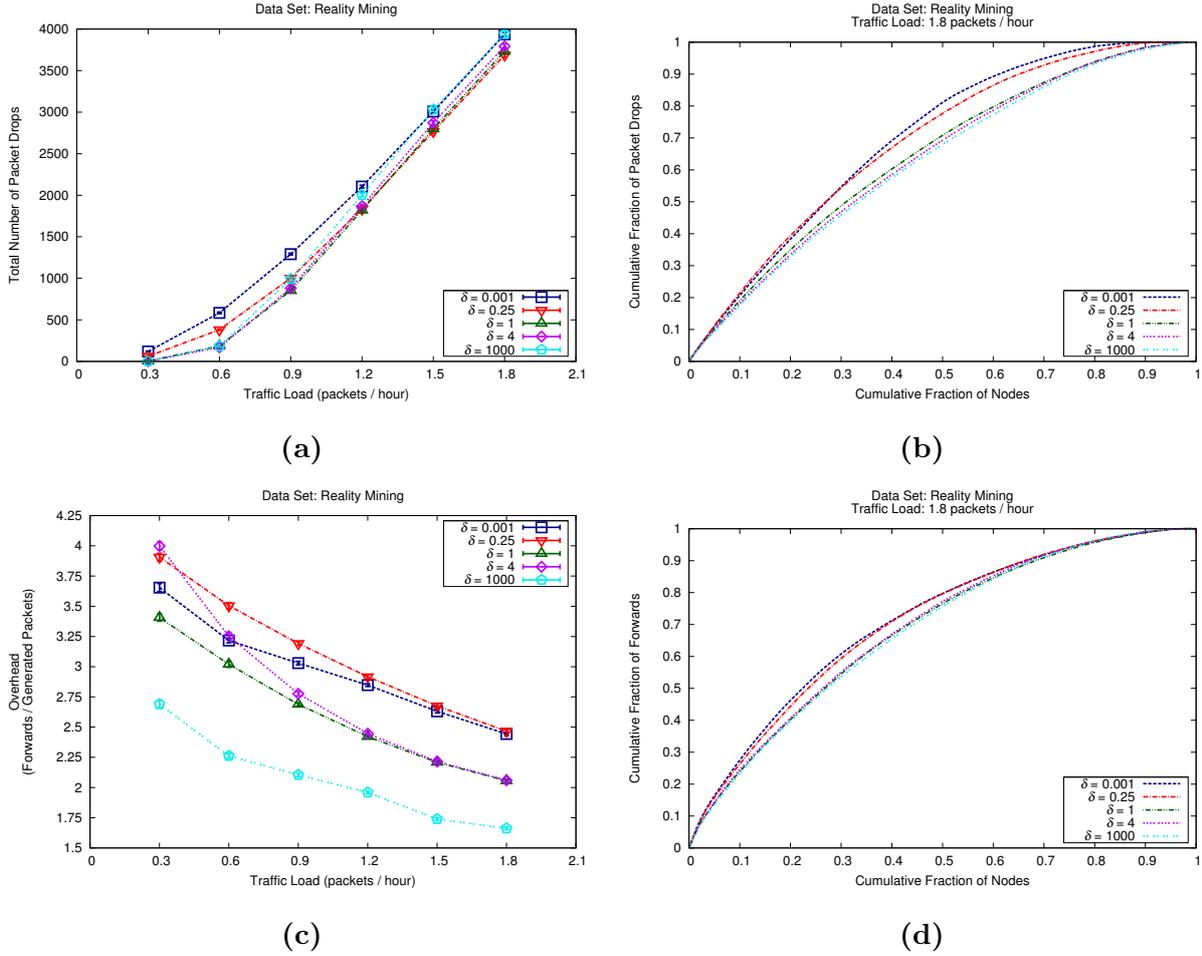
We are going to use the same  $\delta$  values that were used in Figure 3.3 to explain their impact on the forwarding condition. For  $\delta = 1$  we get a linear relationship between the importance of each packet and its effect on the forwarding condition. For  $\delta = 0.25$  we relax the forwarding condition so that we can transmit most of the important packets, while for  $\delta = 4$  we enforce the load balancing between each pair of nodes. Finally, we also examine the two extreme cases, by setting  $\delta = 0.001$  and  $\delta = 1000$ . We will compare our results from simulations on the Reality Mining data set, with the simulation settings that were given in Table 4.2.

As Figure C.1a shows, the approaches with the lowest  $\delta$  values deliver their packets a lot faster, because the nodes seize most of their opportunities to forward their packets to more suitable carriers. On the other hand, the approaches with the highest  $\delta$  values have longer delivery times, because they search for alternative routes that are less congested. As expected, by relaxing the forwarding condition, the packets reach their destinations faster, if they do not get dropped in the mean process due to storage constraints.



**Figure C.1:** The impact of  $\delta$ : (a) Average Delay, (b) Cumulative Distribution Function of the Delay, (c) Delivery Ratio, and (d) Average Number of Hops

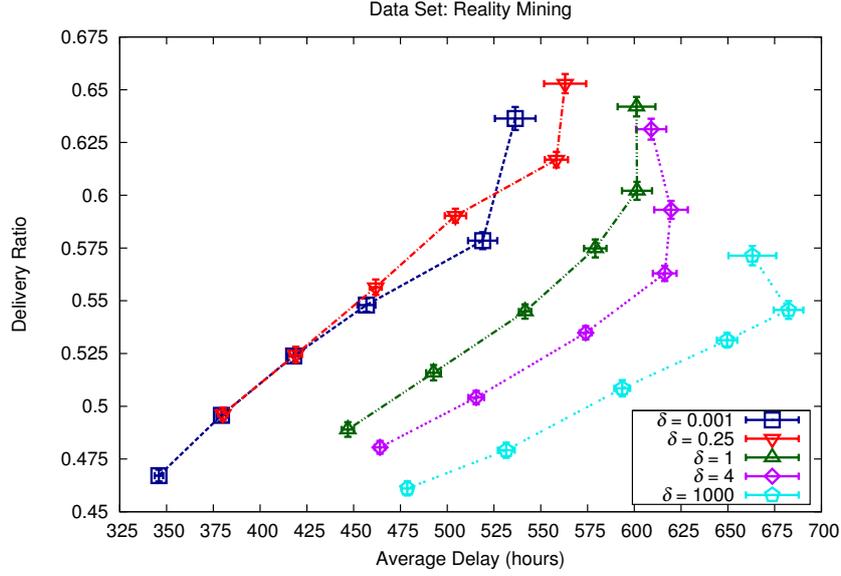
As we can see in Figure C.1b, the approaches with  $\delta$  equal to 0.001 and 1000 were unable to deliver packets that required a long period of time to reach their destinations. This phenomenon is also reflected in the results of the delivery ratio in Figure C.1c. As we can observe, by relaxing the forwarding condition we can increase the amount of successfully delivered packets. However, by setting the  $\delta$  value close to 0 we obtain the opposite results, because each node accepts all the forwarding packets until it runs out of space. As a result, the buffers of the nodes with the highest utility values are most of the time full. Therefore, when they try to generate a new packet, they are forced to drop another one, which causes the reduction in the delivery ratio. On the other hand, by assigning a high  $\delta$  value, most of the opportunities that could help achieve data delivery will be lost to maintain the load balancing among the nodes.



**Figure C.2:** The impact of  $\delta$ : (a) Total Number of Packet Drops, (b) Cumulative Fraction of Packet Drops, (c) Overhead, and (d) Cumulative Fraction of Forwards

By looking at the average number of hops in Figure C.1d, it is clear that the approaches with  $\delta < 1$  deliver their packets usually through the most important nodes. This is the main reason why they are able to deliver their packets faster. On the other hand, the approaches with higher  $\delta$  values often search for alternative paths that are less congested in order to deliver their packets. However, if the traffic load is high, they may not ever find an alternative path that is not under congestion, resulting in packet drops and reduction of the delivery ratio.

As we can observe in Figure C.2a, the approaches with high  $\delta$  values have fewer packet drops under low traffic load, since it is easier to allocate their resources evenly. On the contrary, when the traffic load is high, the approaches with low  $\delta$  values have fewer packet drops, because they reduce the resources that are needed in order to carry all the packets



**Figure C.3:** The impact of  $\delta$  on the Average Delay vs. Delivery Ratio

by delivering them to their destinations. Furthermore, as we can see in Figure C.2b, the approaches with high  $\delta$  values, achieve a nearly uniform distribution of the packet drops, since they focus on distributing the traffic load among the nodes evenly.

Obviously, the approaches with the lowest  $\delta$  values tend to have the least forwards, as shown in Figure C.2c, because of their strict forwarding conditions. However, there are more factors that affect the total number of forwards in the network. For instance, the approach with  $\delta = 0.001$  does not have the most transmissions among the others, because a lot of packet drops occurred due to congestion. On the other hand, the approach with  $\delta = 0.25$  performed more forwards, because it was able to deliver most of the packets with fewer packet drops. As we can see in Figure C.2d, the approaches that have the highest  $\delta$  values, also have the most fair distribution of forwards among the nodes. Interestingly, even the approach with  $\delta = 0.001$  does not result in a heavily skewed distribution.

To summarize, we showed that by sacrificing the fairness in the network to a certain degree, we can improve its performance significantly, without excessive use of the most important nodes. As shown in Figure C.3, when the  $\delta$  parameter has a value less than 1, we can achieve high delivery ratio, as well as low delay. On the other hand, by assigning a value greater than 1 to the  $\delta$  parameter, we can increase the fairness in the network without essentially reducing the delivery ratio.

# Short Vita

Dimitrios-Georgios Akestoridis was born in Athens, Greece in 1991. He graduated from the 26th General Lyceum of Athens in 2009, and in the same year he began his undergraduate studies at the Department of Computer Science and Engineering of the University of Ioannina. He worked as a Network Engineer at Algosystems S.A. during a summer internship, followed by an eight-month voluntary work at the Systems Support Group of the University of Ioannina. Since 2012 he is a member of the Networks Research Group of the University of Ioannina. His research interests lie in the areas of Opportunistic and Delay-Tolerant Networks, Mobility Modeling, and Social Network Analysis.