Δρομολόγηση Σε Κοινονικά Οπορτουνιστικά Δίκτυα Μεγάλης Κλίμακας

Η ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ ΕΞΕΙΔΙΚΕΥΣΗΣ

υποβάλλεται στην

ορισθείσα από την Γενική Συνέλευση Ειδικής Σύνθεσης

του Τμήματος Μηχανικών Η/Υ και Πληροφορικής Εξεταστική Επιτροπή

από τον

Αθανάσιο Κουφούλη

ως μέρος των Υποχρεώσεων για τη λήψη του

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ

ΜΕ ΕΞΕΙΔΙΚΕΥΣΗ

ΣΤΑ ΥΠΟΛΟΓΙΣΤΙΚΑ ΣΥΣΤΗΜΑΤΑ

Ιούνιος 2015

Acknowledgements

At this point i would like to thank all those people who helped, in various ways, in the completion of this thesis.

I want to express my gratitude to Prof. Evangelos Papapetrou for his guidance, supervising and support throughout this research. Additionally, i would like to thank Nikos Papanikos for those endless hours of discussions, which gave me important feedback and various ideas.

Finally, the deepest gratitude to my family, for their support and tolerance all those years.

Table of Contents

1 Introduction **1**

2 Background **4**

2.1 Social Analysis . . . 5

2.2 Network Diameter . . . 6

2.3 Scalling . . . 7

3 CrossWorld: increasing the realism in synthetic traces **10**

3.1 CrossWorld Description . . . 11

3.1.1 Building the high-level topology . . . 12

3.1.2 Generation of contacts between different blocks . . . 13

3.2 CrossWorld Validation . . . 15

4 Routing performance evaluation **18**

4.1 Routing In DTN Networks . . . 18

4.2 Routing Algorithms Description . . . 19

4.3 Simulation Environment . . . 20

4.4 Simulation Results . . . 21

5 Proposed Algorithm **25**

5.1 Ego Network . . . 26

5.1.1 Ego Betweenness . . . 27

5.1.2 Unique Ego Network Nodes . . . 28

5.2 Coordinating Replication . . . 28

5.3 Crossover Description . . . 29

5.3.1 Routing away from Destination . . . 30

5.3.2 Routing near Destination . . . 31

5.3.3 Decreasing number of replication . . . 31

5.4 Crossover Evaluation . . . 31

6 Conclusions **35**

List of Figures

2.1 Fraction of paths with respect to the number of hops they consist of for different traces 6
2.2 Reality Trace 8
2.3 Cambridge Trace 8
2.4 Infocom05 Trace 8
2.5 Evolution of the effective network diameter for the Reality trace 8

3.1 Fraction of paths with respect to the number of hops they consist of for traces produced using SWIM[15]: a) 300 nodes, and b) 500 nodes 11
3.2 Visualized example of CrossWorld output using three clusters based on Reality using Gephi[1] 12
3.3 Pseudocode for contact creation 13
3.4 CCDF of Inter-contact time in real and synthetic traces produced by CrossWorld. a) Real world traces, b) CrossWorld with $p = 0.1, bp = 1$, c) CrossWorld with $p = 0.4, bp = 1$, d) CrossWorld with $p = 0.1, bp = 4$. 15
3.5 Fraction of paths with respect to the number of hops they consist of for traces produced using STORM a) RRR and b) CCCC for various inputs. . . . 16
3.6 CCDF of contact duration in real and synthetic traces produced by CrossWorld.a) Real world traces, b) CrossWorld with $p = 0.1, bp = 1$, c) CrossWorld with $p = 0.4, bp = 1$, d) CrossWorld with $p = 0.1, bp = 4$ 17

4.1 RRR scenario - Performance in terms of delivery efficiency and routing cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$ 21
4.2 RRR scenario - Performance in terms of delivery efficiency and delivery delay cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$ 22
4.3 C4 scenario - Performance in terms of delivery efficiency and routing cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$ 23

4.4 RMC scenario - Performance in terms of delivery efficiency and routing cost (a) extreme case ($p = 0.1$ $bp = 1$) (b) moderate case ($p = 0.4$ $bp = 1$) . 24

5.1 Ego-network & Ego Betweenness example 26
5.2 Coord[21] procedure among u, v at time t 29
5.3 CrossOver procedure among u, v at time t 30
5.4 RRR scenario - Performance in terms of delivery efficiency and routing cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$. 32
5.5 RRR scenario - Performance in terms of delivery efficiency and delivery delay cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$ 33
5.6 C4 scenario - Performance in terms of delivery efficiency and routing cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$. 34

6.1 RRR scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 1$ 40
6.2 RRR scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 2$ 40
6.3 RRR scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 4$ 41
6.4 C4 scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 1$ 41
6.5 C4 scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 2$ 42
6.6 C4 scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 4$ 42
6.7 RMC scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 1$ 43
6.8 RMC scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 2$ 43
6.9 RMC scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 4$ 44
6.10 C4 scenario - Performance in terms of delivery efficiency and routing cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$. 44
6.11 C4 scenario - Performance in terms of delivery efficiency and delivery delay cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$ 45

6.12 C4 scenario - Performance in terms of delivery efficiency and delivery delay cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$ 46

List of Tables

2.1 The studied opportunistic traces 5

3.1 Statistics of CrossWorld traces 14

Abstract

Koufoulis, Athanasios, MSc., Computer Science & Engineering Department, University of Ioannina, Greece. June, 2015. Routing in large scale social opportunistic networks Supervisor: Evangelos Papapetrou.

Opportunistic networks are abstract networks where no infrastructure is required in order to ferry data among nodes. In opportunistic networks, real human trace are widely used for evaluating and tuning routing algorithms designed for these networks. Those traces collected by recording human movement and interaction through wireless devises. The main downside of these datasets is their small scale.

To tackle this problem, recent efforts focus on producing synthetic traces that inherit the properties of real ones. More specifically, so far, the focus is on replicating the distributions of contact duration and inter-contact times in an effort to increase the realism of such traces. All of those models focus in capturing a realistic mobility model for human movement in order to produce contact distribution similar to real human traces. Real human traces as well as proposed synthetic models, exhibit of small degree of separation among nodes which eventually affects routing process, and so, evaluation results is optimistic to be generalized.

We show that it is reasonable to assume an increased degree of separation when considering large scale networks. Since this problem has not been tackled so far, we propose a new paradigm for developing synthetic traces. We do not concentrate on capturing a human mobility model, but use social networks schemes in order to produce traces. We focus on increasing the degree of user separation while at the same time keeping all social characteristics of real human traces.

Additionally, we evaluate and show that state-of-the-art routing algorithm face significant performance challenges in networks with increase degree of separation among nodes, in compare to their results in real human traces.

In order to overcome this performance degradation in routing, we introduce a new algorithm that makes different routing decisions, regardless of its distance with the destination of each packet. Algorithm uses different utility metrics to evaluate nodes, for their ability to deliver a packet and for their ability to spread packet efficiently across the

network, in order to increase successful delivery possibilities.

We evaluate our routing algorithm in newly produced traces, generated by our model, as well as in real human traces and show that this strategy achieves better results in terms of delivery success ratio, transmissions of packets and delay of routing process.

Εκτεταμενη Περιληψη στα Ελληνικα

Κουφούλης Αθανάσιος του Χρήστου και της Ἰννας. MSc, Τμήμα Μηχανικών Η/Υ & Πληροφορικής, Πανεπιστήμιο Ιωαννίνων, Ιούνιος, 2015. Δρομολόγηση Σε Κοινωνικά Οπορτουνιστικά Δίκτυα Μεγάλης Κλίμακας. Επιβλέποντας: Ευάγγελος Παπαπέτρου.

Στο περιβάλλον των οπορτουνιστικών δικτύων μεγάλο ερευνητικό βάρος έχει δοθεί στην μελέτη δεδομένων που προέρχονται από την καταγραφή, πραγματικών, επαφών συσκευών που μεταφέρονται από ανθρώπους. Το μείζον πρόβλημα αυτών των δεδομένων είναι η μικρή τους κλίμακα. Για την αντιμετώπιση αυτού του προβλήματος αρκετά συνθετικά μοντέλα παραγωγής παρόμοιων, μεγάλης κλίμακας, δικτύων έχουν προταθεί με βασική στόχευση, να προσομοιώσουν την χρονική κατανομή της απόστασης μεταξύ των επαφών, καθώς επίσης και της διάρκειάς τους. Κινούμενοι στο προαναφερθέν πλαίσιο και με στόχο την αύξηση της ρεαλιστικότητας, εστιάζουμε σε ακόμη ένα χαρακτηριστικό, την απόσταση μεταξύ των κόμβων σε επίπεδο καταμέτρησης των ενδιαμέσων κόμβων (degree of seperation).

Αρχικά εστιάζουμε στην παραδοχή ότι σε ένα ρεαλιστικό, μεγάλης κλίμακας, οπορτουνιστικό δίκτυο, οι αποστάσεις μεταξύ των κόμβων που θα υφίστανται θα εμφανίζουν μεγαλύτε -ρο πλουραλισμό σε σχέση με τα έως τώρα δεδομένα και ακολούθως, καταδεικνύουμε την αδυναμία των υπαρχόντων συνθετικών μοντέλων να το επιτύχουν.

Για την αντιμετώπιση αυτού του προβλήματος, προτείνουμε ένα νέο μοντέλο, το οποίο βασίζεται στα δεδομένα που έχουν συλλεχτεί από πραγματικά δίκτυα. Το προτεινόμενο μοντέλο, παρέχει παρόμοιες κατανομές σε χρονική απόσταση μεταξύ επαναλαμβανόμενων επαφών και χρονική διάρκεια αυτών σε σχέση με τα πραγματικά δεδομένα. Επιπλέον, διατηρεί και τα κοινωνικά χαρακτηριστικά των προαναφερθέντων δικτύων.

Στη συνέχεια, αξιολογώντας, με ένα σύνολο μετρικών, τους επικρατέστερους αλγορίθμους δρομολόγησης στα οπορτουνιστικά δίκτυα, παρατηρούμε ότι αδυνατούν να επιτύχουν την ίδια απόδοση εν συγκρίσει με τα δεδομένα από πραγματικά δίκτυα.

Αποσκοπώντας στη βελτίωση της αποτελεσματικότητας της δρομολόγησης, εισάγουμε μία νέα στρατηγική δρομολόγησης με κύριο χαρακτηριστικό, τη λήψη ανεξάρτητων αποφάσεων δρομολόγησης ανάλογα με την απόσταση των υπό επαφή κόμβων από τον προορισμό του πακέτου. Συγκρίνοντας, την προτεινόμενη στρατηγική δρομολόγησης παρατηρούμε βελτίωση στην αποτελεσματικότητα της δρομολόγησης στα μεγάλης κλίμακας δίκτυα, συγκρινόμενη με τους επικρατέστερους αλγορίθμους δρομολόγησης.

Chapter 1

Introduction

In opportunistic networks, user contact traces collected in mobile networks are valuable tools not only for evaluating the performance of networking mechanisms but also for designing and fine-tuning them. These traces provide us extensive information about human movement and interaction. A variety of algorithms which has been proposed in resent years focus on taking advantage of those characteristics and make forward decisions based on them. The downside is that most of those traces are of small scale in terms of nodes and some of them in terms of time.

The alternative is to use synthetic traces. The main scope of these traces is to provide larger networks comparing with real traces. Several approaches for creating such traces have been proposed[2], all based on the modeling of user mobility. The algorithms can be classified based on the type of user mobility, e.g., human, vehicular, etc. In the case of human mobility, which is the scope of our job, the common ground in all approaches is to model mobility as a result of human activities and social ties. In an effort to increase the realism of such synthetic traces, recent approaches have focused on inheriting certain characteristics that appear in real traces[13],[15]. More specifically, the traces produced by the latter methods have been shown to exhibit a power law with an exponential decay dichotomy distribution of inter-contact times, similar to the one observed in real traces[14], as well as a similar distribution of contact duration. These characteristics are important because they explore the forwarding opportunities providing for data transfer among nodes.

Although this line of research moves towards the right direction, we make the observation that there are other important characteristics of real human traces that should be examined when producing large scale synthetic traces. Motivated by research efforts that reveal the "small world" phenomenon when examining the shortest path in real human traces[4], [20], [3], we focus on a slightly different structural characteristic such as the *degree of separation* between nodes, i.e. the minimum number of hops required for delivering a message from one node to another.

In Section 2 we show that real human traces exhibit a very small degree of separation which can be clearly attributed to the small scale of the experimental networks used to collect them. Therefore, it is reasonable to assume that larger networks will exhibit a larger degree of separation. Unfortunately, current synthetic mobility models are not able to model such a behavior. We prove this observation by analyzing several outputs of these models.

To tackle this problem, we propose a new paradigm for producing large scale synthetic traces. Our approach, which we call *CrossWorld*, does not rely on modeling user mobility. Instead, we use real human traces as building blocks and then model the interaction of users from different blocks. This way, we are able to model a higher degree of separation while, at the same time preserve certain characteristics observed in real traces such as the distribution of inter-contact times, the distribution of contact duration and the clustering coefficient. We demonstrate *CrossWorld* model in Section 3.

Additionally, in Section 4 we evaluate state-of-the-art routing algorithms in traces generated of our model. These algorithms seem to achieve a much lower performance in a set of evaluating metrics compared to real world human contact traces. The main reason for this degradation is the differences in network's topology, such the degree of separation among most of source destination pairs. The evaluation takes place in different topologies and sizes of networks. The results seem to validate our observation that these algorithms which were designed to perform in traces where nodes are separated by a small amount of intermediate nodes.

In Section 5 we try to investigate the main reasons of this degradation and purpose a new algorithm which attempts to overcome the new challenging environment in a sufficient way. Our algorithm achieves better performance in these topologies without experiencing low performance in real world human contact traces. This is done by evaluating network's topology in a more sophisticated way in order to make forwarding decisions. In order to achieve that, we use different utilities for replication depending on our distance from destination node, and a explore network topology to identify bridge nodes. We finally, compare CrossOver with state-of-the-art algorithms and demonstrate the results for both synthetic traces produced with CrossWorld and real human contact traces.
Summarizing, our contributions are:

- We show that real human traces exhibit a very small degree of separation and we argue that, when producing large scale synthetic traces, it is reasonable to assume an increased degree of separation (Section 2).

- We propose *CrossWorld*, a new paradigm for constructing synthetic traces from real ones (Section 3). CrossWorld is able to produce synthetic contact traces that exhibit a configurable degree of separation and a series of characteristics observed in real human traces.
- We show that state-of-the-art routing algorithms exhibit a systematic performance degradation when the degree of separation increases (Section 4).
- We demonstrate an algorithm to cope with new network topology and we evaluate in both synthetic and real world traces(Section 5)

We conclude this work in Section 6.

Chapter 2

Background

2.1 Social Analysis

2.2 Network Diameter

2.3 Scaling

Opportunistic networks are abstract networks where no infrastructure is required in order to ferry data among nodes. Recent years, the vast majority of research efforts in opportunistic networks has been driven by human contact traces, usually referred as Pocket Switch Networks (PSN). Those traces has been collected by recording human interaction and movement under specific scenarios like university camps or conferences Table 2.1. Participants were carrying devices which exchange data by using wireless protocols such bluetooth or wifi. A contact exists when two or more devices come in range. By recording these contacts over time, a graph of forwarding opportunities is constructed. Due to their disrupting nature the forwarding scheme defers from the traditional. Intermediate nodes store & carry data before forwarding them, taking advantage of the next contact opportunity. A path between a source and destination pair exists when a chain of contact opportunities connecting those two nodes exists.

Because of the scenario collected, those traces reveal several information about people movement and interaction. A big amount of research focused on model the movement pattern and other in statistically analyzing those traces. One of the most important findings is that in human contact traces exhibit a series of "small world" properties[][][]. In advance, they have been widely used for evaluating routing algorithms designed for forwarding this kind of networks. Such algorithms try to explore those networks topology and take advance of their characteristics. Despite their useful aspect of recording human

Table 2.1: The studied opportunistic traces

Trace Name	# Nodes	Duration (days)	Technology	Area
Infocom '05[25]	41	≈3	Bluetooth	conference
Infocom '06[25]	98	≈4	Bluetooth	conference
Sigcomm '09[22]	76	≈3.7	Bluetooth	conference
MIT Reality[7]	97	≈283	Bluetooth	campus
Milano pmtr[19]	44	≈19	Bluetooth	campus
Cambridge [16]	36	≈11.4	Bluetooth	campus

interaction, their main downside is that they are of small scale. The alternative is to use synthetic generated traces. Several approaches for creating such traces have been proposed[1], all based on the modeling of user mobility. The algorithms can be classified based on the type of user mobility, e.g. human, vehicular, etc. In the case of human mobility, the common ground in all approaches is to model mobility as a result of human activities and social ties.

In this section we will present the findings of that statistical analysis and make an intuition about how these traces would scale.

2.1 Social Analysis

Human contact traces include a variety of information about human movement and relations. Most of those information can be mined by investigating the contact between individuals. It is not quite obvious, how these wireless contacts, produced by human movement and interaction can be model in social relations, as refereed in social networks theory. These contacts are recorded when two wireless devices, carried by humans, are in range. This connection does not implicates relationship between two individuals. For example, they could just be in the same place like a bar.

In [14], authors have focused in statistically analyze those traces in terms of contact distribution. The statistical analysis revealed certain properties that provide useful hints on how contacts are distributed over time. Authors has shown that the time which mediates among two contacts between a pair of nodes follows power-law decay up to a characteristic time value, and then follows an exponential decay. They also provide informations about intra-contact durations and contact repetition distribution. These findings are important because they conclude the forwarding opportunities that arise in this kind of networks.

In [11] authors investigate the social structures between participants in these complex

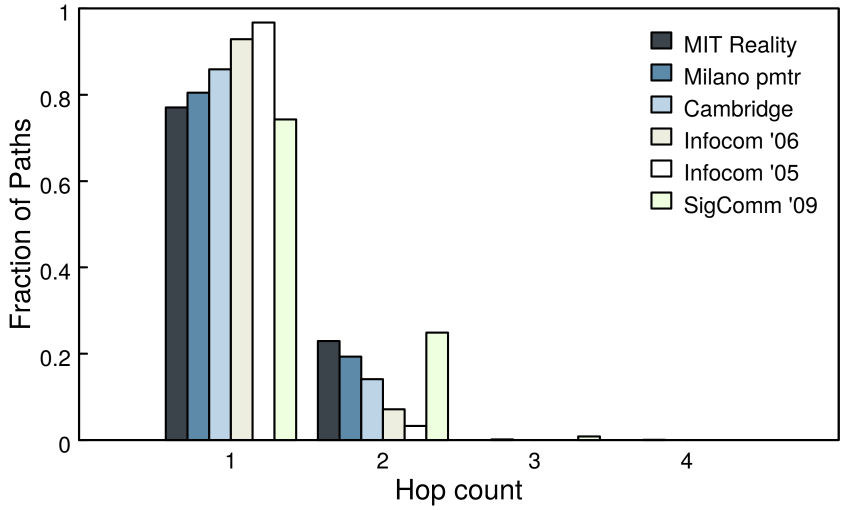


Figure 2.1: Fraction of paths with respect to the number of hops they consist of for different traces

networks. Authors show that different relationships between nodes can be found by evaluating their contacts in terms of frequency, duration, and repetition. Some contacts represent social relationship, while others seem to be random. They also present that using this kind of analysis we can found that further social structures exist. Community cluster can be found in those traces as well as further social behavior. Members of these clusters are connected in a higher rate than those who are not. This kind of analysis is quite useful for boosting routing performance. Although, random links seem to be quite useful for delivering messages in these networks. In [24] show that network connectivity among these traces depends not only in frequent contacts but also in rare ones. This happens because these relatively rare contacts reduce delay of packet delivery and also provide further connections between nodes in different community clusters.

All these findings about the social characteristics of these networks triggered further analysis about the distribution of paths and other small world properties like the small diameter of the network.

2.2 Network Diameter

In [4],[3] authors suggests that the diameter of such a network equals to four. In order to prove that argument they concentrated in the optimal path appeared between a pair, starting from a random time and found that their distance is small. More specifically, using different methods, researchers have focused on statistically analyzing the hop count of the "optimal path" for a source-destination pair[4],[3]. Different interpretations of the term “optimal”, such as *fastest*[4],[3] or *most probable*[3], have been examined. In all cases, it was found that paths with more than four hops do not provide significant performance improvements.

Although the aforementioned strategy is reasonable for providing useful insight into the routing process, it is questionable whether it can capture the structural characteristics of the network such as the actual *degree of separation* between nodes. The reason is that the analysis considers only the connectivity provided by the “optimal” paths. However, it is well-known that for each source-destination pair there is a multitude of paths[8]. In this work, authors studied the *path explosion* i.e. large amount of new paths appearance, which occurs shortly after the fastest path between a pair of nodes appear. Therefore capturing the degree of node separation, e.g. for deciding on the network's scaling properties, requires taking into account all available paths and examining the *minimum hop count* experienced by each source-destination pair. In this context, we expect the degree of node separation to be much smaller than four. This is because of the delay-hop count trade-off in human contact traces[3] and time-varying graphs[17] in general. In general, fastest path seems to be longer than paths that appear shortly after that, due to previous mentioned path explosion. If such a finding is confirmed, a reasonable question is:
Q: Will larger networks of this type continue to exhibit such a low separation degree?

2.3 Scalling

To validate our observation, we statistically analyzed the hop count of the minimum hop path in various traces (Table 2.1). For the analysis, we examined the minimum hop paths for all possible pairs using the corresponding time-varying graph (TVG)[31], i.e., a graph where each edge is labeled with a presence function, since TVGs are known to produce more accurate results[31],[3]. Fig. 2.1 illustrates the fraction of minimum hop paths versus the number of hops they contain for all examined traces. For all traces the plethora of paths contains at maximum two hops, i.e., when starting from a node all other nodes can be reached within two hops. Only in two traces, Milano pmtr and Sigcomm, there is a negligible percentage of paths that consist of more than two hops ($\approx 0.2\%$ for the Milano $\approx 0.8\%$ for the Sigcomm trace). In all traces, most paths consist of only one hop. Furthermore, for most traces the effective diameter[17], i.e., the 90th percentile of all paths distance, is two while for Infocom'05 and Infocom'06 it is one. Our findings are in accordance with some sparse results obtained for a subset of the aforementioned traces using a different methodology[3].

Then we want to monitor how effective diameter evolves over time. Since, in this type

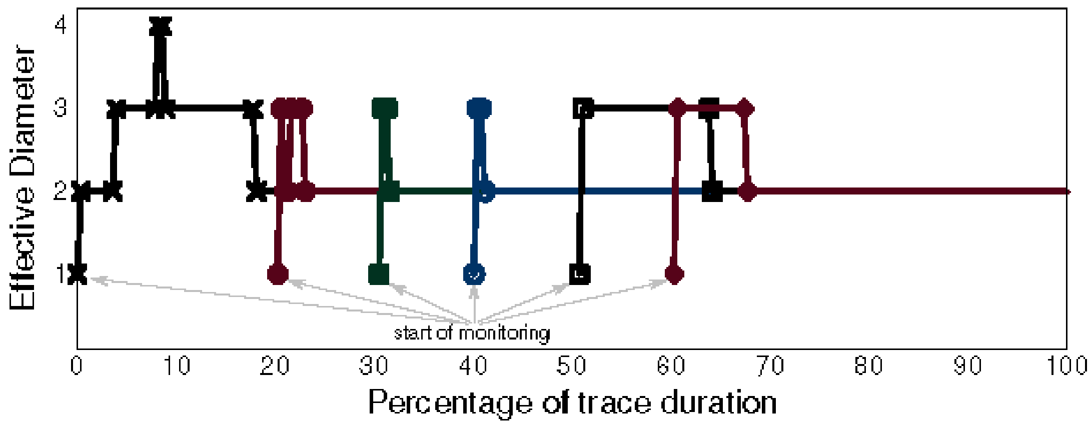


Figure 2.2: Reality Trace

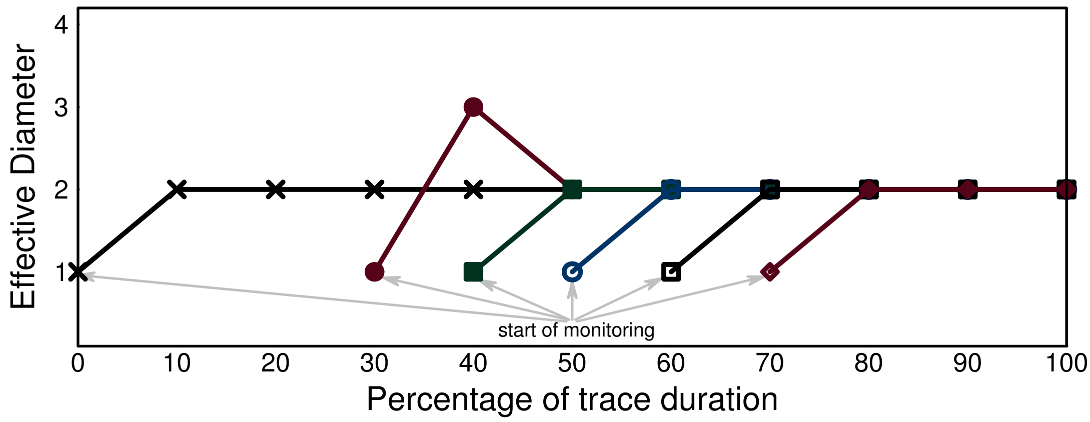


Figure 2.3: Cambridge Trace

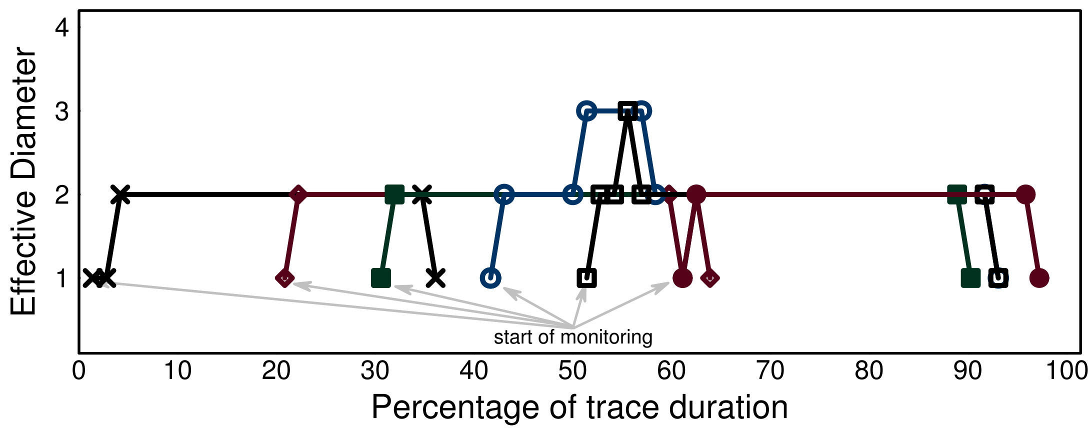


Figure 2.4: Infocom05 Trace

Figure 2.5: Evolution of the effective network diameter for the Reality trace

of networks paths do not exist but only for a short period of time, is important to validate this observation in time. To accomplish this we write down the distance from every node to other starting from different times and recording them every few hours. The interesting finding is that the effective diameter quickly converges to its minimum value regardless of the time we start monitoring the network. Fig. 2.5 illustrates the evolution of the effective diameter over time for a variety of traces. Different lines correspond to different starting times for the monitoring process. In all cases, the diameter quickly converges to its minimum, two for Reality and Cambridge and one for Infocom05 Trace, indicating that this is a persistent and not a transient characteristic of the network. The small prolongation of the convergence period witnessed, for some traces, in the beginning and towards the end of the trace duration is due to nodes entering and leaving the network. We obtained similar results for all other traces. Clearly, all reported results reveal a very small degree of separation which is reasonable considering the type of networks used to collect the aforementioned traces. In all cases the experiment: a) takes place in a limited geographic area (e.g., conference area, campus), and b) the participants are related at least by a "loose" relationship (e.g., some type of enrollment in the same campus, participants in the same conference). Since it is rather unlikely that this will be the case with larger networks, we argue that the answer to question (Q1) is that *the scaling of such networks will involve an increased degree of separation between nodes.*

Naturally, this brings forward two important questions:

Q2: Is an increased degree of separation going to affect the performance of state-of-the-art routing algorithms?

Q3: How to construct synthetic traces featuring a higher degree of separation?

A first response to Q2 is yes if we keep in mind the correlation of the degree of separation with the hop count of the shortest path. We examine in detail this issue in Section 4. As for Q3, since current synthetic models have never looked into this type of scaling, it is not clear whether they can provide a solution. We resume with this issue in Section 3.

Chapter 3

CrossWorld: increasing the realism in synthetic traces

3.1 CrossWorld Description

3.2 CrossWorld Validation

So far, the algorithms for producing large scale synthetic traces have focused on replicating the distribution of inter-contact times[14] and the distribution of contact duration observed in real human traces, in an effort to enhance the realism of the produced traces. To this end, they model typical mobility patterns observed in real traces such as users frequently visiting specific points of interest[13] or even visiting some points of interest more often than others[15] Since this approach replicates the user behaviour observed in real traces, it is reasonable to assume that the degree of separation will not increase regardless of the number of users in the network. To validate this observation, we statistically analyzed a set of large scale traces produced using the SWIM model[15]. We used three different real traces (Cambridge, Infocom'05 and Infocom'06) as the reference trace and produced traces for 300 and 500 nodes. We used the Phoenix model for expanding the network since it results in more hops[15]. The results (Figure 3.1) confirm that the degree of separation does not increase regardless of the number of nodes in the network.

Additionally, Despite those models seem to achieve similar behaviour in inter-contacts duration and produce an analogous behaviour in contact duration distribution, it is not quite obvious that they keep all the social characteristics, like clustering coefficient and group formulation, with real traces [29]. In this work authors investigate the similarity between nodes i.e. the common neighbours between a pair of investigated nodes and come to a conclusion that till today proposed synthetic trace models are incapable of producing

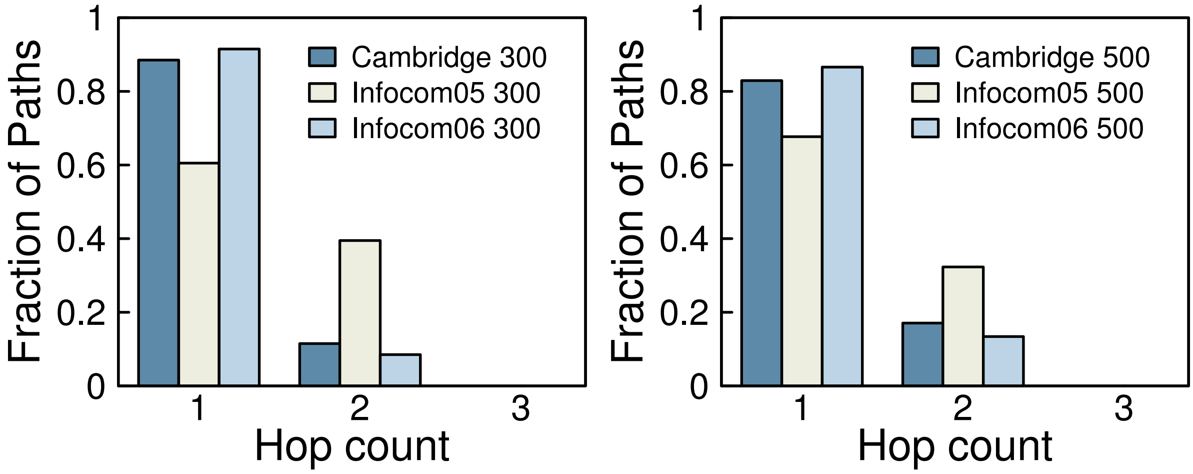


Figure 3.1: Fraction of paths with respect to the number of hops they consist of for traces produced using SWIM[15]: a) 300 nodes, and b) 500 nodes

different similarities among users of the network and do not achieve high clustering coefficient. On the other hand, this are important aspects of real world traces and some routing algorithms rely on them in order to make forwarding decisions. In this work, we take a different approach. Instead of modeling user mobility, we visualize small scale real traces as the building blocks of a larger network. Each block represents a network that evolves in a specific geographic area and its users are related at least by a "loose" relationship, e.g. people enrolled in a campus. Note that this does not rule out the existence of user groups with "tighter" relationships within such networks. In such a scenario, some users have multiple enrollments in different small scale networks, being in this way the "glue" that brings together the building blocks. We believe that this paradigm for creating synthetic traces, which we call "CrossWorld", provides a high-level view of a real-life large scale social mobile network. We will validate our intuition in Section 3.2 by showing that "CrossWorld" is able to create synthetic traces that:

- exhibit a more realistic degree of separation between users
- inherit several characteristics of real traces such as the clustering coefficient, the distribution of inter-contact times as well as the distribution of contact duration

First, we describe "CrossWorld" in detail.

3.1 CrossWorld Description

As mentioned previously, the key concept in CrossWorld is to model large scale networks as a collection of smaller ones. Users may enroll in more than one of the smaller networks,

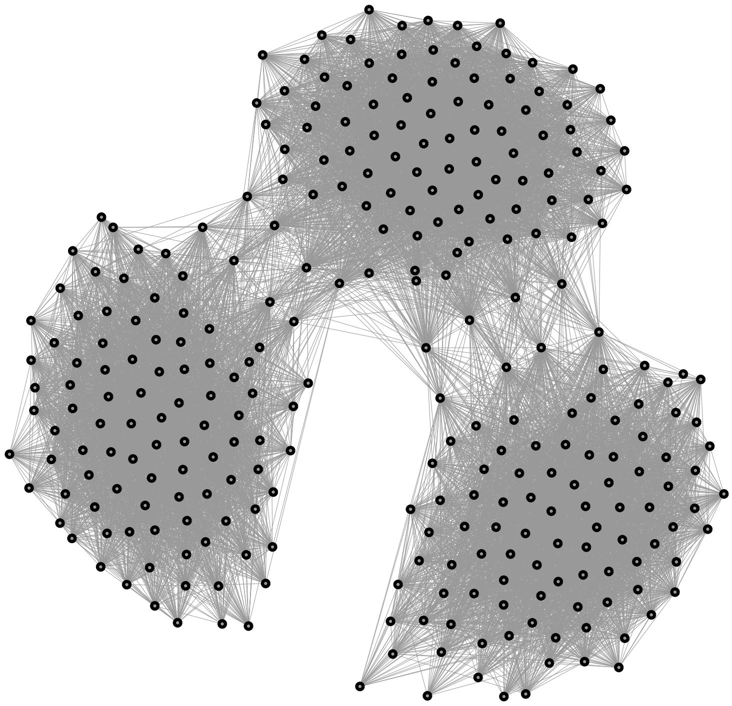


Figure 3.2: Visualized example of CrossWorld output using three clusters based on Reality using Gephi[1]

providing the connectivity between the building blocks. Our model consists of two main steps:

1. Building the high-level topology
2. Generation of contacts between different blocks

3.1.1 Building the high-level topology

The first step is to choose the number of building blocks. As a building block we can use any type of real human contact trace (e.g. campus, conference) or any other synthetic trace (e.g. SWIM[15]). Note that there is no requirement that the building blocks are of the same type. For example, it is possible to use a real trace as one block and a synthetic one as another. The next step is to establish "connections" in the high-level topology. By the term "high level topology" we mean a topology where the building blocks are represented as nodes while a connection between two blocks translates in a number of contacts between nodes of the two blocks.

INPUT: $cnodes(u)$: nodes that u encounters
$clist(n, u)$: contacts between nodes n and u
t_{bound}: bound in contact time variation
OUTPUT: CSet: set that contains the new contacts

```
function CreateNewContacts(v, u)
    for every node n ∈ cnodes(u) do
        if rand(0,1)¿p then
            for every contact c ∈ clist(n, u) do
                t_rand ← rand(0, t_bound)
                t_start ← c.t_start ± t_rand
                t_end ← c.t_end ± t_rand
                C Set ← CSet + contact(v, n, t_start, t_end)
            end for
        end if
    end for
    return CSet
end function
```

Figure 3.3: Pseudocode for contact creation

3.1.2 Generation of contacts between different blocks

After determining the connectivity between the blocks in the high-level topology, the next step is to create the contacts between the nodes of the connected blocks. The model for doing so is inspired from the triadic closure phenomenon, widely observed in social networks[23]. This principle indicates that, it is likely for two people who both have a relationship with the same person to develop a relationship between themselves as well. So we not only need to choose a pair (u, v) of users and connect them. To have a valid model, we also need to produce contacts between v and some of the users that interact with u. Based on the aforementioned principles, Leskovec et al.[17] proposed an algorithm, named Forest Fire, for producing a larger network from a given social network. When a new node v is added in the network, the algorithm selects an existing node u and connects v to the neighbors of u with probability p.

Since human contact traces exhibit social characteristics, the Forest Fire model is a good candidate for producing contacts between blocks in the context of CrossWorld. However, our hypothesis is that even if most of the nodes in a block tend to interact with other blocks, only a subset of them would produce a big amount of consistent contacts.

Table 3.1: Statistics of CrossWorld traces

Trace	bp	p	Clustering Coefficient	Effective Diameter
Reality	-	-	0.840	2
RRR	1	0.1	0.815	5
		0.2	0.813	4
		0.4	0.812	4
	4	0.1	0.804	4
		0.2	0.798	4
		0.4	0.790	3
Cambridge	-	-	0.892	2
C4	1	0.1	0.876	6
		0.2	0.868	5
		0.4	0.860	5
	4	0.1	0.829	4
		0.2	0.804	4
		0.4	0.781	4

We call these nodes the "bridge nodes" and implement the Forest Fire algorithm only for those nodes. This means that for each bridge node v we randomly select a node u in the neighbour block. Then, we decide with probability p whether to create a contact between v and w, for each w which is an encounter of u. The contacts that will be produced between v and w, will have the same distribution of start and end time, as the contact between u and w. We add a small variation to the start and end times in order to avoid starting all contacts in the exact same time. In addition, we perform the same operation for creating contacts in both directions for each pair of connected blocks. The algorithm of this procedure is described in Fig. 3.3. The number of bridges bp is a parameter of the algorithm that allows us to establish different levels of connectivity between the different blocks. Note that the set of nodes that interact between two blocks is a superset of the set of bridge nodes. Fig. 3.2 visualizes a CrossWorld trace with three blocks, each one based on MIT Reality, and $(bp, p) = (4, 0.4)$.

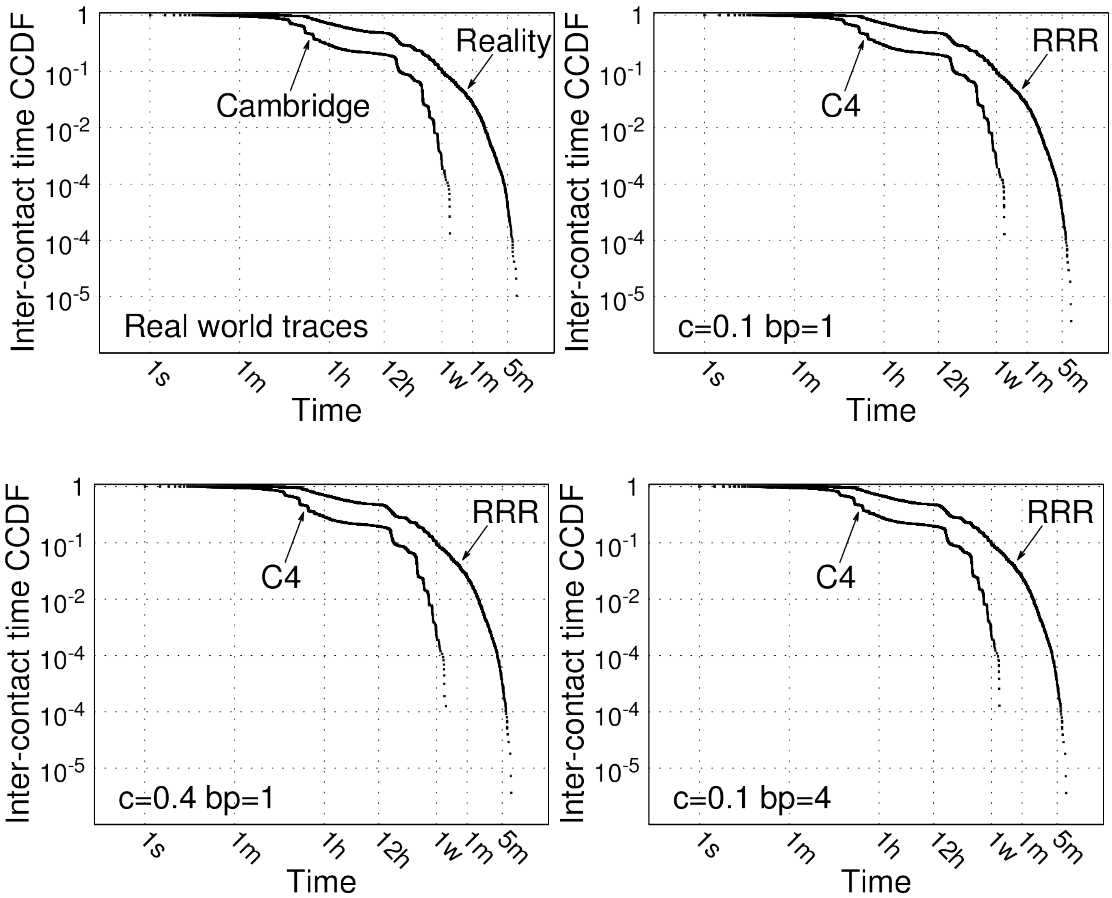


Figure 3.4: CCDF of Inter-contact time in real and synthetic traces produced by CrossWorld. a) Real world traces, b) CrossWorld with $p = 0.1, bp = 1$, c) CrossWorld with $p = 0.4, bp = 1$, d) CrossWorld with $p = 0.1, bp = 4$.

3.2 CrossWorld Validation

As mentioned previously, the primary objective of CrossWorld is to produce synthetic traces that exhibit a higher degree of separation compared to real traces. The degree of separation can be configured through the number of blocks and parameters bp and p. Moreover, in order for the CrossWorld traces to be more realistic, it is desired to exhibit a set of attributes that have been witnessed in real traces. Such attributes are:

- a high clustering coefficient [29],
- a power law distribution of inter-contact time with an exponential decay dichotomy[14].
- finally, achieving a contact duration distribution similar to that of real traces would further enhance the realism of CrossWorld.

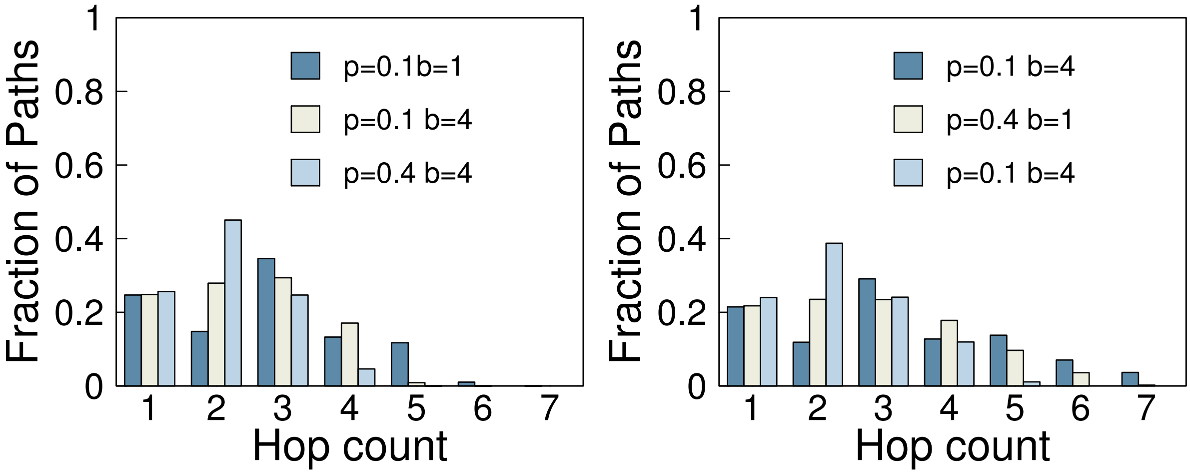


Figure 3.5: Fraction of paths with respect to the number of hops they consist of for traces produced using STORM a) RRR and b) CCCC for various inputs.

To evaluate the performance of CrossWorld with respect to the aforementioned objectives, we tested different combinations of building blocks, and various values for bp and p. For brevity we present two representative examples. In the first, we used the Reality trace as the building block with a total of three blocks. We will refer to this example as RRR. In the second example, we used four blocks with the Cambridge trace as the building block. We will refer to this example as C4. Table 3.1 presents the effective diameter when considering the minimum hop paths as well as the clustering coefficient for the example traces and for different values of bp and p. It is clear that it is possible to configure the degree of separation using parameters bp and p. As p increases the distance between nodes is shrinking. The same applies when we increase bp. Both results are reasonable since more connections between the building blocks are created. Note that different (bp,p) value pairs may result in the same effective diameter, however the distribution of minimum hop paths according to their hop count (Fig. 3.5) varies. Finally, in Fig. 3.4 we demonstrate the CCDF of inter-contact time for Reality, Cambridge as well as for RRR and C4 with $bp=1$,$p=0.1$. The power law with the exponential decay dichotomy is inherited by both RRR and C4. Moreover, the distribution is virtually identical to that of Reality and Cambridge, respectively. We also found that this holds for every value of bp and p. This is a reasonable result considering our policy for creating contacts. For the same reason, the same resemblance appears when studying the distribution of contact duration.

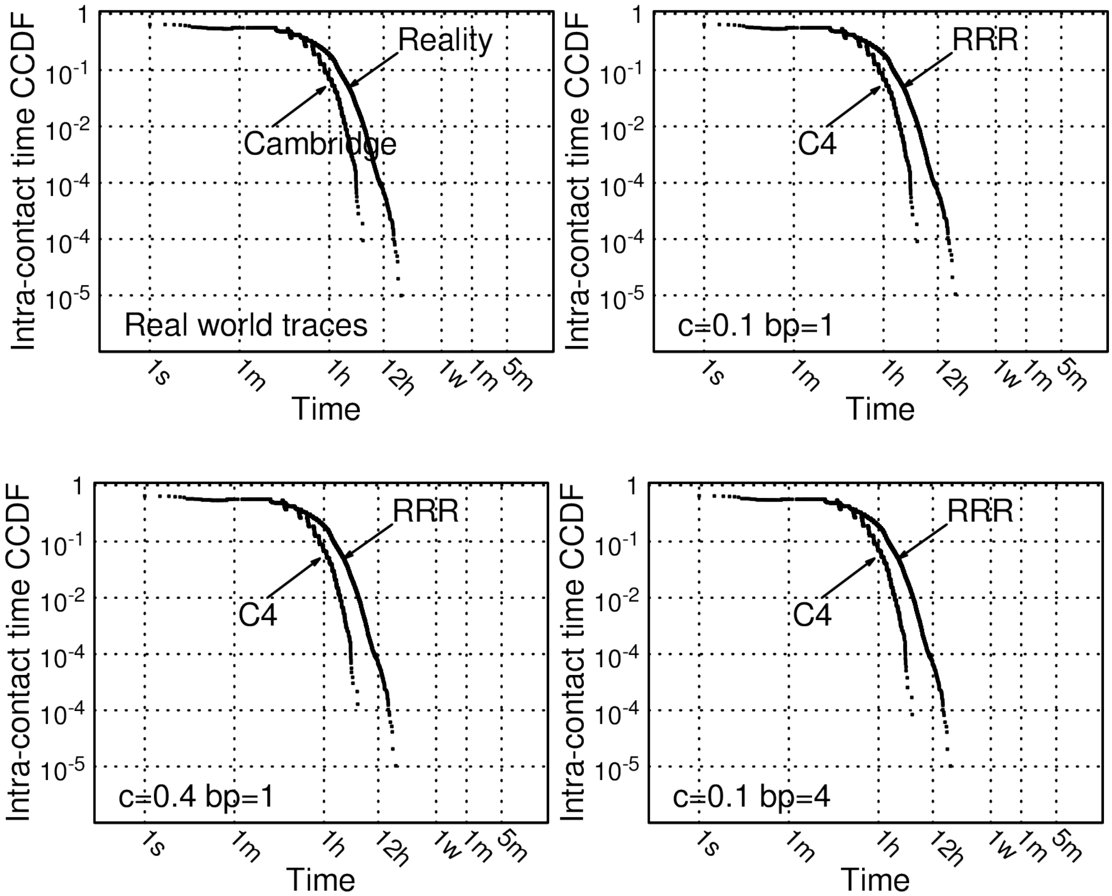


Figure 3.6: CCDF of contact duration in real and synthetic traces produced by CrossWorld.a) Real world traces, b) CrossWorld with $p = 0.1, bp = 1$, c) CrossWorld with $p = 0.4, bp = 1$, d) CrossWorld with $p = 0.1, bp = 4$

Chapter 4

Routing performance evaluation

4.1 Routing In DTN Networks

4.2 Routing Algorithms Description

4.3 Simulation Environment

4.4 Routing Algorithms Evaluation

In this section we examine question (Q3), i.e., what is the impact of an increased degree of separation to the performance of state-of-the-art routing algorithms. To this end, we developed a custom event-driven simulator that operates on a contact basis and is able to integrate the synthetic traces generated by our model as well as real human traces. In the experimental study we used the synthetic traces RRR, C4 and a heterogeneous RMC, i.e. Reality-Milano-Cambridge being the building blocks. We used various values of p and bp.

4.1 Routing In DTN Networks

In general, routing in this kind of networks follows the store-carry-forward scheme due to the disruption nature of the network and lack of instantaneous source to destination paths. This means that in a contact between a pair of nodes $u\&v$, where u forwards a packet to v, node v stores it until another transfer opportunity arises.

Two main strategies has been proposed for forwarding in DTN, Single-copy and Multi-copy. In single-copy strategy, only one replica of the forwarding packet exists every single time in the network. Each node who forwards a packet then deletes it from his buffer. In multi-copy strategy the idea is two have more than one replica in the network at the same time in order to increase delivery probability. Thus, node replicates instead of forwarding a message two an encounter. The disadvantage of this method is that it increases the cost of network resources needed.

There exists two prime approaches for producing replicas in multi-copy strategy. The first uses a predefined L number of replicas and spreads it among users, till is reached. The main disadvantage is that L must be predefined and it has a big impact on algorithm's delivery probability. Most of times it is not possible to determine the number of replicas, a big number increases network resources needed, and a small may The second approach is to replicate if encounter node has better probabilities for delivering message than carrying node. This comparison is done by several utilities (e.g. Last time each node saw destination[28]) that evaluate node's ability. In this approach, some authors [21][9], produced schemes in order to decrease the number of replications.

Utilities are separated in two categories. Destination dependent utility and destination independent. Destination depended utilities are evaluating a nodes ability to deliver a message to a certain destination while independent evaluates node significance for the whole network.

4.2 Routing Algorithms Description

We selected five well-known algorithms, namely Epidemic, BubbleRap, PRoPHET, SimBetTS and Spray & Focus. We will now give a brief description of each algorithm.
Epidemic[30]: This protocol floods the network with message replicas. Every node replicates every message when a contact opportunity arises. Thus, it accomplishes the highest achievable performance in terms of delivery rate and message delay. On the other hand this protocol is the upper bound in transmissions since every node could have a replica of a message.
BubbleRap[12]: BubbleRap distributively ranks all network nodes based on their popularity in the entire network (global ranking), as well as, their popularity inside their communities (local ranking). Routing is accomplished using the global ranking scheme until a node in the same community with the destination is discovered. From that point the local ranking scheme is enabled to reach the destination.
PRoPHET[10]: PRoPHET exploits the past encounter history in order to predict the probability of future encounters, a.k.a. the delivery predictability. It computes the probability of every node to deliver a message and makes forward decisions by comparing this probability.
SimBetTS[5] SimBetTS uses multiple social based metrics that are locally estimated at

each node. The fundamental metrics are similarity and ego betweenness, while the other metrics are tie strength indicators that measure how strong or weak is the relationship among network nodes. Routing is accomplished by replicating a predefined number of message replicas to the encountered nodes with higher SimBetTS metric.

Spray & Focus[27]: Spray & Focus consists of two phases: the spray phase, where a predefined number of message replicas are disseminated in a greedy manner. Every node copies to encounter the half of the amount of the number of messages that it has. Message replicas are spread in network binary. This face ends when a node has only one replica and so it is unable to copy. Then, the focus phase starts. From this point, nodes with single-copy messages forward them based on a utility metric. In our study, we use the LTS[28] metric that is calculated as $1/(1 + \text{LastTime})$, where LastTime is the elapsed time since the last contact with the destination.

4.3 Simulation Environment

After conducting extensive experimentation, we concluded that algorithms using a single-copy strategy fail to compete with those using a multi-copy strategy. For that reason and in order to test the very essence of the routing logic of each algorithm we adopted a multi-copy strategy for all protocols. Furthermore, we implement the vaccine deletion scheme [26] for cleaning up redundant replicas after successful delivery. For algorithms that use a predefined number of message replicas L, such as SimBetTS and Spray & Focus, we present the performance for an optimally chosen L value, i.e., the value that results in the best performance. For all protocols we use the parametrization recommended by the authors.

In order to clearly capture the impact of separation we choose to generate traffic only between nodes that reside in the most distant groups. In accordance with most evaluation studies in the literature, we avoid packet drops due to congestion by setting each node to have unlimited storage capacity. Randomness is introduced both in the production of the synthetic datasets and the traffic generation. We randomly generate 1000 packets in the interval during which both the source and the destination are present in the network. Furthermore, we use a warm-up and a cool-down period, during which packets are not generated. The duration of each period is 20% of the total trace duration. The reported results are obtained as the average of 50 repetitions. In all cases, we present the confidence interval with a 95% confidence level.

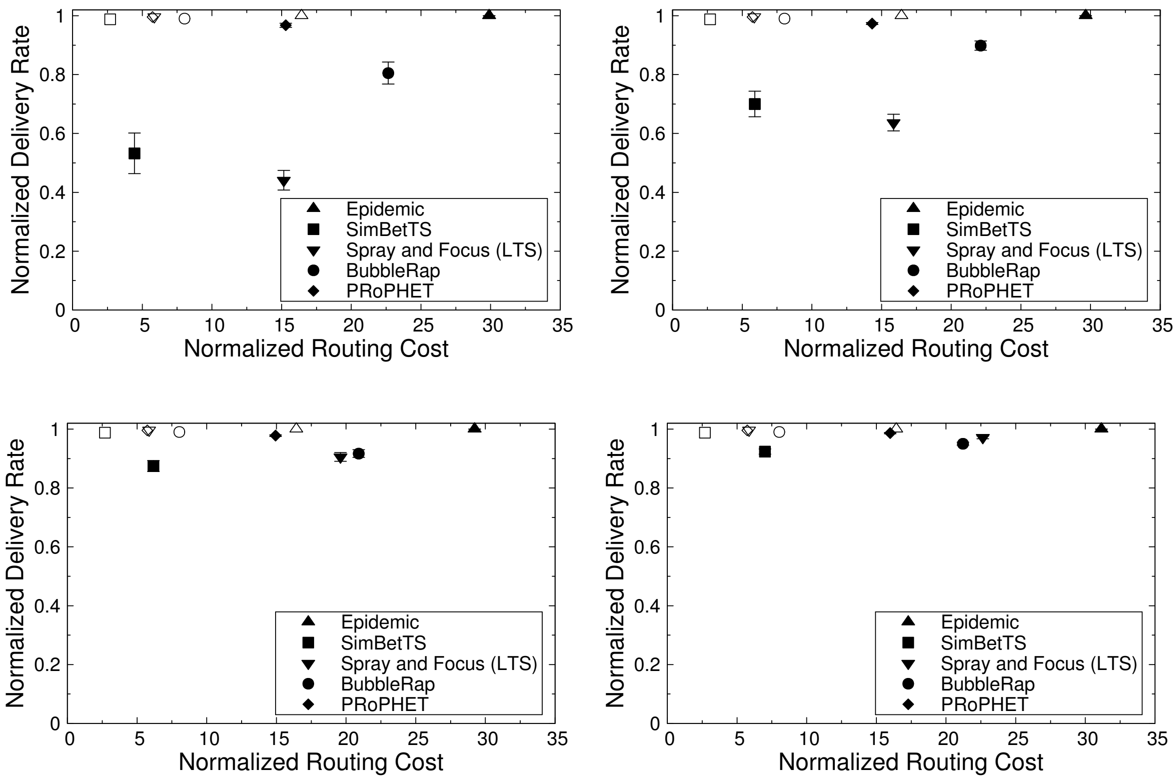


Figure 4.1: RRR scenario - Performance in terms of delivery efficiency and routing cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$

4.4 Simulation Results

To explore the performance of the investigated protocols we use the following three metrics: the delivery rate, the routing cost (i.e. the total number of transmissions) and the average message delay. In order to enable the comparison of routing performance in CrossWorld and real traces, we present each of the above metrics normalized to the optimal performance. More specifically, we developed an optimal algorithm that exploits the full knowledge about the network topology to discover the minimum hop path between the set of fastest paths for each generated message. Through this normalization we capture the distance of each algorithm from the optimal performance in the corresponding trace. Note that the Epidemic algorithm achieves the optimal performance in terms of delivery rate and delay, therefore its normalized performance equals to one. Fig. 4.1 depicts the performance of all protocols in the RRR case for different values of bp and p. We use filled points to represent the performance of each protocol in the CrossWorld traces, while the non-filled ones represent the corresponding performance in the Reality dataset. Note that a point closer to the upper left corner indicates a performance closer the optimal one. As expected, the performance of all protocols is highly correlated to the degree of

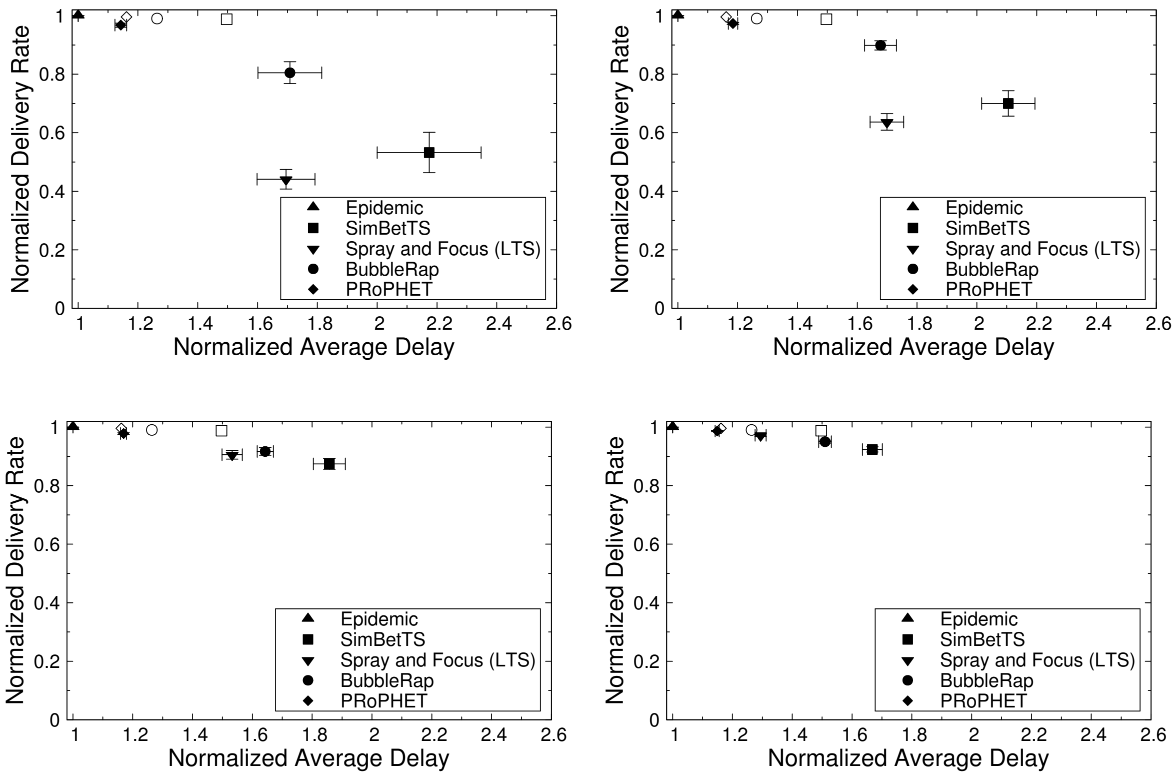


Figure 4.2: RRR scenario - Performance in terms of delivery efficiency and delivery delay cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$

separation between the network nodes. More specifically, in the case of $p = 0.1$,$bp = 1$ (Fig. 4.1(a), 4.2(a)), which according to Table 3.1 represents an extreme case of separation, all algorithms suffer from a severe degradation in their performance. This degradation appears either as poor delivery efficiency (SimBetTS, Spray & Focus) or excessive routing cost (PRoPHET, BubbleRap). For example, the delivery ability of SimBetTS decreases by half while the transmissions produced by PRoPHET increase by $\sim 300\%$ compared to the transmissions of PRoPHET in the MIT Reality trace. The same holds for the delay performance of all algorithms with the exception of PRoPHET (Fig. 4.2(a)). Increasing either the link creation probability p (Fig. 4.1(b), 4.2(b)) or the number of bridges bp (Fig. 4.1(d), 4.2(d)) improves the performance of all algorithms. This is reasonable since according to Table 3.1 the degree of separation is reduced. The performance improvement is more noticeable when bp increases instead of p. This is because more paths consist of less hops (Fig. 3.2). Nevertheless, in both cases, every algorithm presents a degraded performance with respect to at least one of the metrics. For example, although PRoPHET manages a good delivery ratio and average delay, its performance in terms of routing cost is significantly lacking. Similarly, while SimBetTS manages an improved delivery ratio, the average delay is still far from optimal. Further increasing either p or bp reduces the

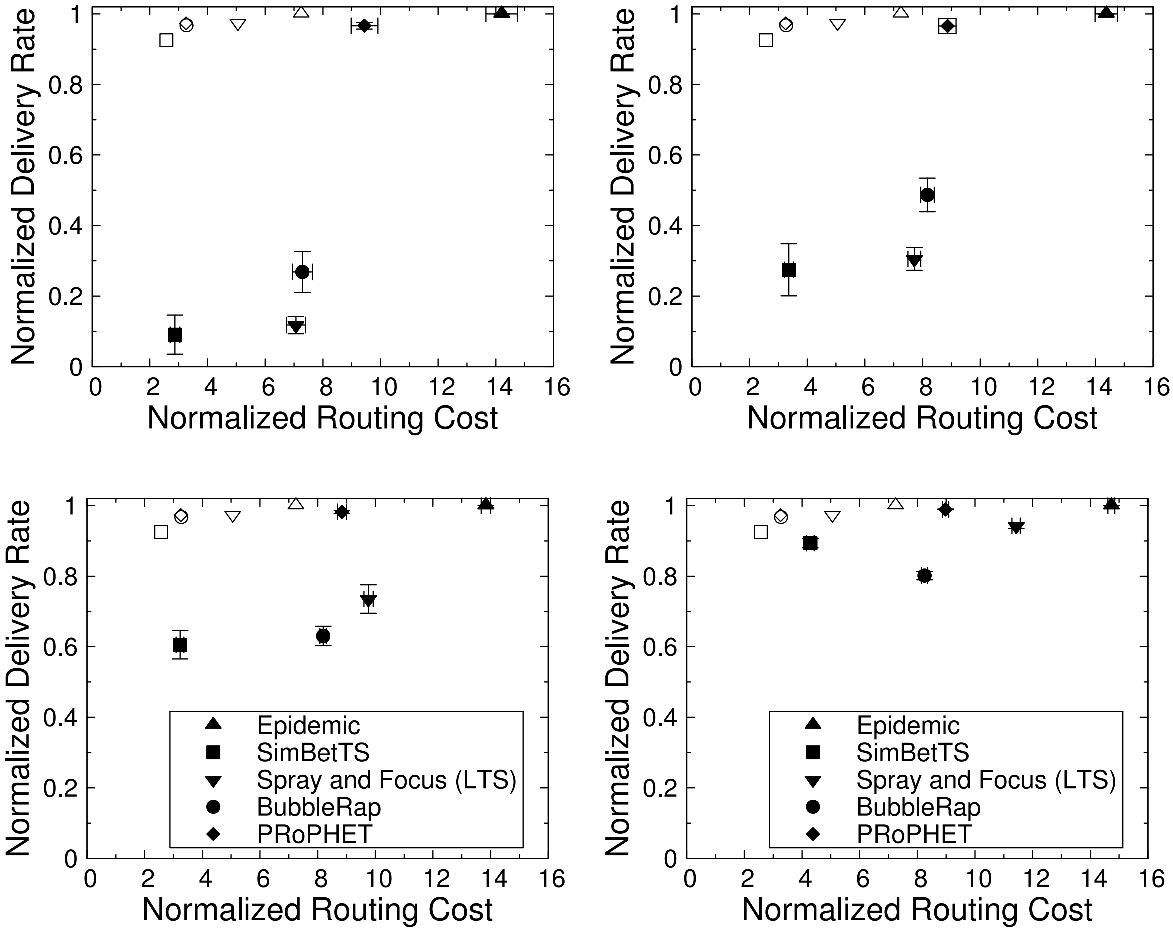


Figure 4.3: C4 scenario - Performance in terms of delivery efficiency and routing cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$

degree of separation therefore improves the performance of all protocols.

Fig. 4.3 illustrates the performance of protocols in the C4 synthetic trace. As a first observation, C4 represents an even more challenging topology since the effective diameter ranges from 4 to 6 (Table 3.1). This is evident in the performance of all algorithms which is significantly degraded compared to the RRR case. PROPHET still manages the best delivery ratio, however its routing cost is very high ($\sim 64\%$ of Epidemic's cost). Again, increasing either bp or p results in performance improvement for all algorithms due to the a smaller effective diameter.

Fig. 4.4 illustrates the performance of all protocols in the RMC case under varying degrees of separation between the network nodes. RMC is a heterogeneous synthetic trace composed of three different real world datasets, i.e., MIT Reality, Milano pmtr and Cambridge. As a reference point, we also display the corresponding performance of all algorithms under the MIT Reality trace (non-filled points in Fig. 4.4). This real

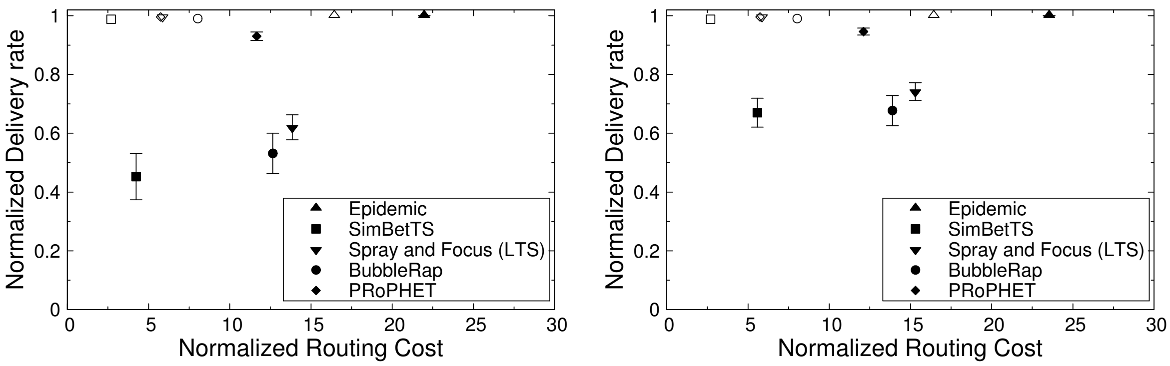


Figure 4.4: RMC scenario - Performance in terms of delivery efficiency and routing cost (a) extreme case ($p = 0.1$ $bp = 1$) (b) moderate case ($p = 0.4$ $bp = 1$)

world trace is the most demanding in terms of routing cost compared to the other two used to produce RMC. Again, similar to the RRR case, the same correlation between the performance of all algorithms and the degree of separation among network nodes appears. More specifically, as depicted in Fig. 4.4(a), high degree of separation results in a significant reduction in the delivery efficiency for all algorithms that is up to ~50%. PRoPHET stands as the only exception achieving a competitive delivery performance. Yet, this comes at the expense of excessive routing cost which doubles compared to that in the MIT Reality trace. As expected, when more links are added (Fig. 4.4(b)), all protocols slightly improve their delivery ability and reduce their routing cost. However, their overall performance remains far from that in corresponding the real world traces.

Chapter 5

Proposed Algorithm

5.1 Ego Network

5.2 Coordinating Replication

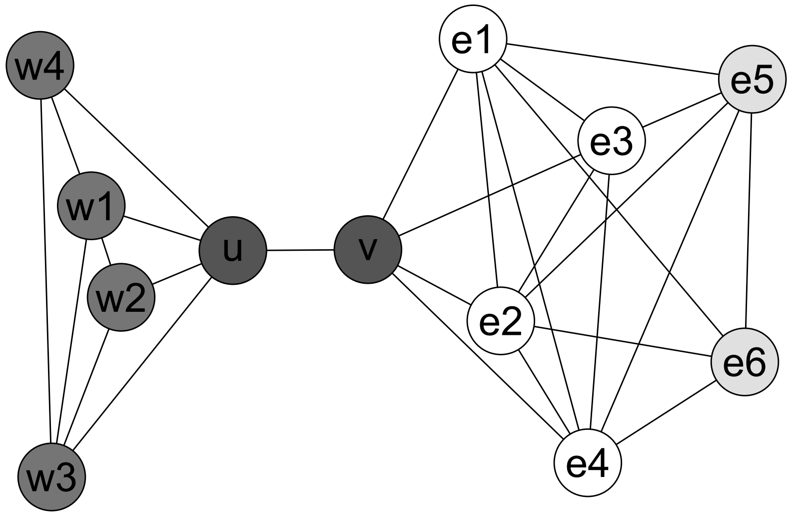
5.3 Crossover Description

5.4 Algorithm Evaluation

In Chapter 4 we demonstrate that state-of-the-art routing algorithm suffering a major degradation in their performance in CrossWorld produced traces. The main reason of this degradation is that these routing algorithms were designed to perform in a smaller degree of separation and though they could not achieve same results in a larger scale network.

Our intuition is that an algorithm which will perform well in a scaled network such CrossWorld should take different forwarding decisions when the encounters are "close" to destination and when not, and though it should use a destination dependent along with an independent metric. Additionally it should be adaptive and avoid inputs i.e. predefined number of replicas of a packet across network. Additionally, single copy approaches fail to achieve good delivery ratio, as mentioned in Chapter 4, and though we follow a multi-copy strategy for routing. The rest of this Chapter relies as follows:

- In [5.1] we will give a short background about ego-network, and rooting decisions that could be taken by mining information of it.
- In [5.2] we will describe Coordination, a replications strategy focusing in decreasing the number of packet replicas in the network.



Node	Ego Bet.
u	4.33
w1	0
w2	0
w3	0.33
w4	0.33
v	4
e1	0.83
e2	0.83
e3	0.25
e4	0.83

Figure 5.1: Ego-network & Ego Betweenness example

- In [5.3] we will describe CrossOver, an algorithm designed to achieve good performance regardless network's topology,
- and finally in[5.4] we will evaluate its routing performance.

5.1 Ego Network

Due to the distributed nature of these networks, every node has limited knowledge about network' topology. In contrast with other kind of networks, and because of the limited resources of mobile devices(e.g. battery, buffer), nodes are incapable to transmit and store whole network's topology. Thus, every node is limited to have knowledge about his short "neighborhood", usually referred as *Ego-network*.

Ego Network usually consist of a node's encounters and the contacts between them. The depth of one-hop is selected because users are needed to exchange information only about their direct encounters, decreasing the necessary transmission. Up to one-hop every node has fully knowledge of the sub-network at time t, i.e. he knows all relations between nodes part of it. An example of ego network is given in Fig. 5.1.

The figure depicts the ego-network of u and v nodes. Part of u's ego network are only the grey nodes and the edges between them. Despite its knowledge about white nodes, they are not consider part of its ego network since u is incapable of knowing information about their relationships. Ego network has significant importance in evaluating nodes, especially by social utilities. Some of them focus in evaluating nodes and their importance over the whole network while other over their relations with a specific node. At this point, we will describe two utilities *Ego Betweenness* and *Unique Ego Network Nodes* which we will use in Sec 5.3 in order to make routing decisions, by evaluating nodes fore their importance over network.

5.1.1 Ego Betweenness

By examining ego network, some researchers proposed centrality measures in order to evaluate nodes importance over the whole network[6], with the most prominent being Ego-Betweenness Centrality. In this work authors showed that a correlation among Betweenness and Ego Betweenness Centrality exists. **Betweenness Calculation** The calculation of the metric is equal to centralized version. For node u, Betweenness Centrality Bet_u is: all shortest paths between every pair of the ego network that includes u in their per all shortest paths of network between this pair.

$$Bet_{u_{sd}} = \sum_{s \neq u \neq d} = \frac{\sigma_{sd}(u)}{\sigma_{sd}},$$

where σ_{sd} is the total number of paths between s, d and $\sigma_{sd}(u)$ the subset of those paths that pass through node u. Note that, a node in most cases in not capable of calculating another node's Ego-Betweenness, due to incomplete knowledge of other's node network i.e. every node knows only a nodes encounters, and not his encounters relation. A comprehensive example of Betweenness Calculation is demonstrated in 5.1

Ego Betweenness is an important utility for evaluating a user, for his importance over the network. A node with high Ego Betweenness is important for its local subnetwork, being for example a bridge node among two *building blocks*. This metric is used by some algorithms [5] to make forwarding decisions. The main problem of such a strategy is that, although it manages to evaluate a node importance quite accurate, it is not easy to overcome the occasions, where a node is a local maximum of Betweenness. To illustrate this, consider the case of Fig. 5.1 that we have two bridge nodes in contact u and v. They both have high values in terms of betweenness centrality but u value is bigger. The destination relies in the cluster that v is in contact ($e6$). Using only betweenness in order to make forwarding decisions we could never reach destination of the packet. In order to overtake this problem, we should evaluate a node's importance even further.

5.1.2 Unique Ego Network Nodes

As we defined previously, ego network is the one-hop distance neighbors of a node. The construction of this network is made by the exchange information about each nodes encounters. As Fig. 5.1 depicts, ue has knowledge about the white nodes and their relation with its encounters. This information may be useful for evaluating a nodes capability for reaching nodes that an encounter could not have any contact without it being the intermediate.

Definition 1 $Uniq_u, (v)$ is the total number of nodes that u can reach only through v. In Fig. 5.1, $e1 - 4$ are Unique nodes for node u, so $Uniq_u, (v) = 4$. If we examine nodes v subnetwork, we found that except u, for which $Uniq_v, (u) = 4$, for all other $Uniq_v, (e_{1-4}) = 0$. For example, e6 despite is not direct contact is not Unique for either of e1 or e2 or e4, since if any of those node would not provide him, some of the others would.

Unique nodes of ego network are an indication of an encounters importance over the network, regarding to node's knowledge over it. It evaluates v about its capability to spread a replica of a packet in subsets of the network that u could not reach otherwise. We introduce this node's evaluation in order to cope with the problem represented in 5.1.1. An advantage of this method is that it does not require any more control packets to be exchanged among u and v.

5.2 Coordinating Replication

Multi-copy strategies have been broadly used for routing in opportunistic networks. The main idea behind using multiple copies of the same packet is to increase the probability of successful delivery to it's destination. Although, it significantly increases delivery ratio and decreases delay of routing, it comes with a cost of increment of routing cost. Two main strategies has been proposed in order to decrease the number of replicas:

- to predefined the number of replicas up to value L
- to use a threshold U_{thresh} and replicate only to nodes that $U_n > U_{thresh}$

The disadvantage of the first approach is that the definition of the best value of L is not straightforward. In most of cases it is selected after extensive experimentation, which can not be done in a real time network. Additionally, for every source-destination pair, the need of replicas differs, and in large scale networks this difference is even more

for every packet $p \in Buf_u$ **do**
    if $p \in Buf_v$ **and** $c\tau^p_{u,t} < c\tau^p_{v,t}$ **then**
        $c\tau^p_{u,t} \leftarrow c\tau^p_{v,t}$
    else
        if $c\tau^p_{u,t} < U_v$ **then**
            Forward p to node v
            $c\tau^p_{u,t} \leftarrow U_v$
            $c\tau^p_{v,t} \leftarrow U_v$
        end if
    end if
end for

Figure 5.2: Coord[21] procedure among u, v at time t

intensive. Furthermore, in large scale networks, the degree of separation among users differs greatly, which makes the right choose of this value even more difficult.

For those reasons, we will choose the second strategy due to its adaptive style. In [21], authors proposed a method of decreasing replication with little impact on delivery ratio and routing delay. The idea behind this approach is that we will replicate a packet to a node only if it is the best candidate from nodes perspective. In more detail, when nodes u and v meets, u will replicate packet p to v if and only if v has achieves the best utility evaluation value among all nodes that u knows that they have the packet in their buffer until meeting time.

Following this scheme, we achieve to replicate a packet only to nodes that are better than those who already have it, with respect to nodes knowledge. To achieve this knowledge every node exchange information with others about his knowledge for the best holder of each packet. In 5.2 is described the decision process among two nodes updating thresholds for this packet $c\tau^p_{u,t}$ at that time t. **Definition 2** $Coord^p_{u,v}$ as the procedure in which u decides to replicate p to u and both update their thresholds $c\tau$

5.3 Crossover Description

I this section we will demonstrate CrossOver, a new routing algorithm, designed to perform in high lever regardless of network topology. In order to achieve this goal we will use two different approaches for routing decisions. We will use different utility evaluation for replication when a node is near to destination and when is not. Additionally, since we follow a multi-copy strategy we use COORD method in order to decrease the number of

for every packet $p \in Buf_u$ **do**
    if $p \in Buf_v$ **and** $dd\tau^p_{u,t} < dd\tau^p_{v,t}$ **then**
        $dd\tau^p_{u,t} \leftarrow dd\tau^p_{v,t}$
        $di\tau^p_{u,t} \leftarrow di\tau^p_{v,t}$
    else
        if $dd\tau^p_{u,t} < U_v$ **then**
            Forward p to node v
            $dd\tau^p_{u,t} \leftarrow U^p_v$
            if $Bet_u < Bet_v$ **or** $Uniq_u, (v) > 0$ **then**
                $di\tau^p_{v,t} \leftarrow U^p_v$
            end if
        else if $(Bet_u < Bet_v$ **or** $Uniq_u, (v) > 0)$ **and**($dd\tau^p_{u,t} \leq U^p_v$) **then**
            Forward p to node v
            $di\tau^p_{v,t} \leftarrow U^p_v$
        end if
    end if
end for

Figure 5.3: CrossOver procedure among u, v at time t

packets replicas across the network. In more detail:

- we use Betweenness Centrality (Bet_u) and Unique Ego Network Nodes ($Uniq_u, v$) evaluation described in Sec. 5.1.1 as destination independent utility,
- we use Last Time Seen [28] as destination dependent utility (U^d_u) with d being the destination.

5.3.1 Routing away from Destination

In large scale DTN it rather usual that some of the nodes would never meet each other for a while. In such a case we are enable to evaluate nodes only in terms of their ability to meet destination since they may never do. To cope with this problem we will evaluate nodes in terms of their importance in the network.

We will use Bet in order to find critical nodes for the subnetwork and $Uniq$ in order to find bridge nodes. In Chapter 2 we demonstrated that in most of datasets the distance between nodes is two, and since the networks diameter converts quickly, for every 2-hop node there are more than one paths. Following this intuition we will copy a message to every node that provides at least one unique node in our ego network.

5.3.2 Routing near Destination

We selected LTS to be our destination dependent utility because it achieves the best performance in most of real world traces. Although, we could use any other from proposed utilities [18] [8] [6]. When two nodes u and v meet, we consider that a node is near to destination if $U_u^p > 0$ or $U_v^p > 0$. In case of LTS any of nodes must have encountered destination.

5.3.3 Decreasing number of replication

In order two decrease number of replication we will use two thresholds, one for destination dependent routing $(dd\tau_{v,t}^p)$ and one for destination independent $(di\tau_{v,t}^p)$, in order to select always the best nodes. For $(dd\tau_{v,t}^p)$ the procedure equals to that of Coord. We exchange information and keep $(dd\tau_{v,t}^p)$ updated to the biggest known value of U^p to every node.

On the other hand, keeping the same approach for $(di\tau_{v,t}^p)$ does not provide the desired effects. Firstly, we do not want to replicate only to nodes that provide the higher known $Uniq$ nodes and additionally as mentioned before we want to avoid the local maximum problems, especially when we are not close to destination. Another argument to avoid using threshold in destination independent utilities, and especially Betweenness Centrality, is that in most cases, its value decreases over time since more and more nodes come in contact.

On the contrary, we want to inform replica holders of a packet, that the packet is close to destination. Therefore, we will use as $(di\tau_{v,t}^p)$ again U^p but we will update it when we replicate to a node that is evaluated as better for the network. The main advantage of this approach is that when we meet nodes that know destination we will inform other nodes and the U^p will start getting high values. In that case nodes that are not close to the destination will stop replicating to others since there will not be nodes with $U_v^p \geq di\tau_{v,t}^p$. The whole procedure is demonstrated in 5.3.

5.4 Crossover Evaluation

In this Section we will evaluate CrossOver in compare to two algorithms who achieve better performance, SimBetTS that achieves good performance with less transmissions than other, PRoPHET which achieves best delivery among others, and Epidemic which is upper bound in delivery ratio and delivery delay. We will follow the same simulation environment as described in 4. We will generate traffic only between nodes that reside in the most distant groups, in the same three metrics. We randomly generate 1000 packets in

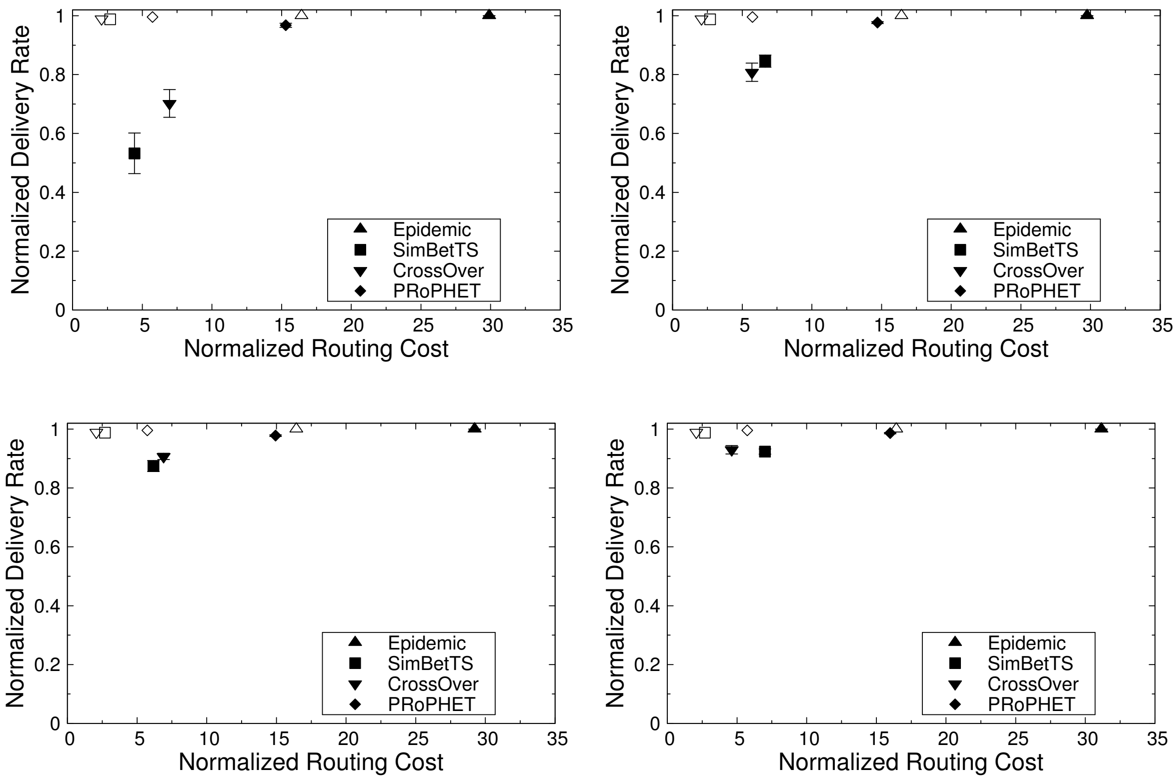


Figure 5.4: RRR scenario - Performance in terms of delivery efficiency and routing cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$

the interval during which both the source and the destination are present in the network.

In Fig. 5.4 we demonstrate results or performance in RRR case. As in 4, filled points to represent the performance of each protocol in the CrossWorld traces, while the non-filled ones represent the corresponding performance in the Reality dataset. In Reality Trace, CrossOver algorithm achieves the best performance in terms of delivery-cost trade-off. I RRR achieves equal or better performance with SimBetTS for all values of bp and p, PRoPHET, still achieves better delivery but with cost of $\sim 300\%$ more transmissions in every case.

In harder case 5.4(a) CrossOver's delivery success is $\sim 70\%$ of optimal and its cost is $\sim 20\%$ of Epidemic's. As expected, when degree of separation becomes smaller all algorithms achieve better performance and in easier case 5.4(d) CrossOver achieves performance close to that of real world traces, both in terms of delivery success and routing cost.

Fig. 5.5 depicts delivery delay of successful delivered packets, normalized with shortest possible delay. Again PRoPHET achieves best performance, but with huge cost in terms of transmissions5.4. CrossOver achieves better performance delay than SimbetTS and as routing becomes easier5.5(d), CrossOver can be compared even with PRoPHET, which

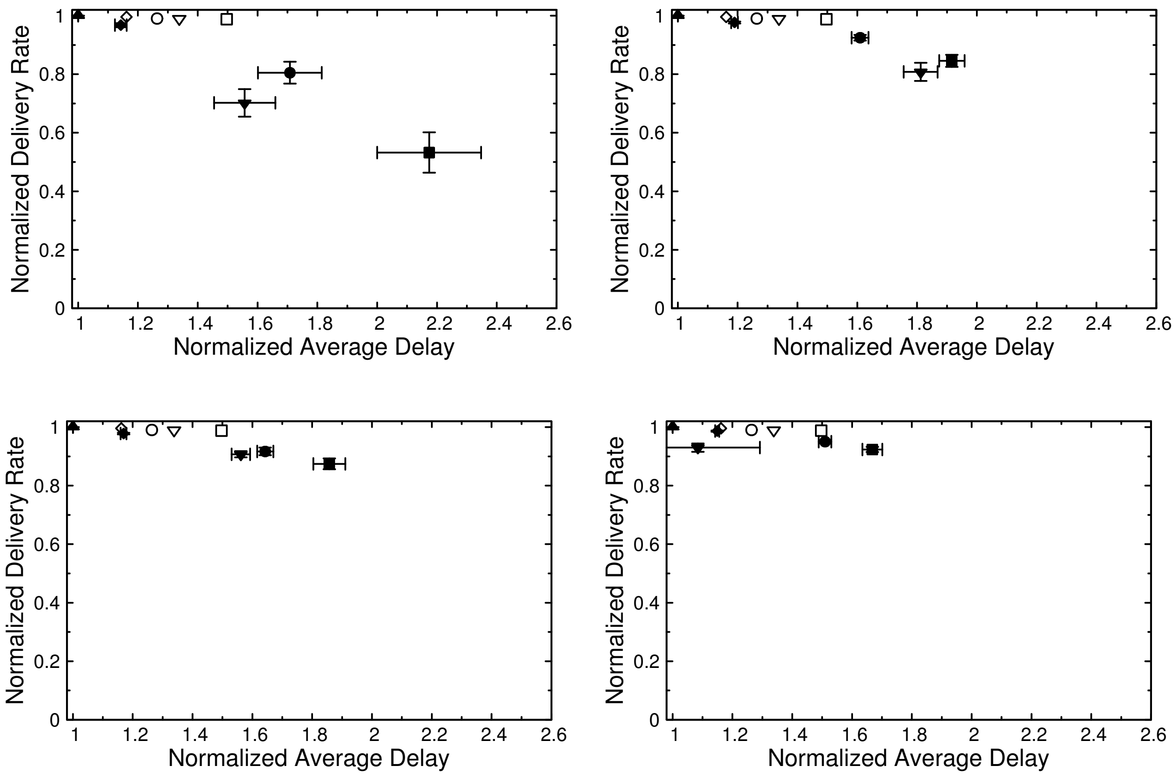


Figure 5.5: RRR scenario - Performance in terms of delivery efficiency and delivery delay cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$

cost is 3 times larger.

C4 5.6 is the most demanding of all datasets we use for evaluation. Again PRoPHET achieves better delivery performance but with high routing cost. When we increase values of bp and p, CrossOver remains better than SimbetTS and with less transmissions. In 5.6(b) 5.6(c) and 5.6(d) CrossOver accomplishes similar delivery performance with half cost, compared to PRoPHET. Additionally, in real world trace (Cambridge) CrossOver produces less transmissions than any other while it manages high performance regarding to delivery success ratio.

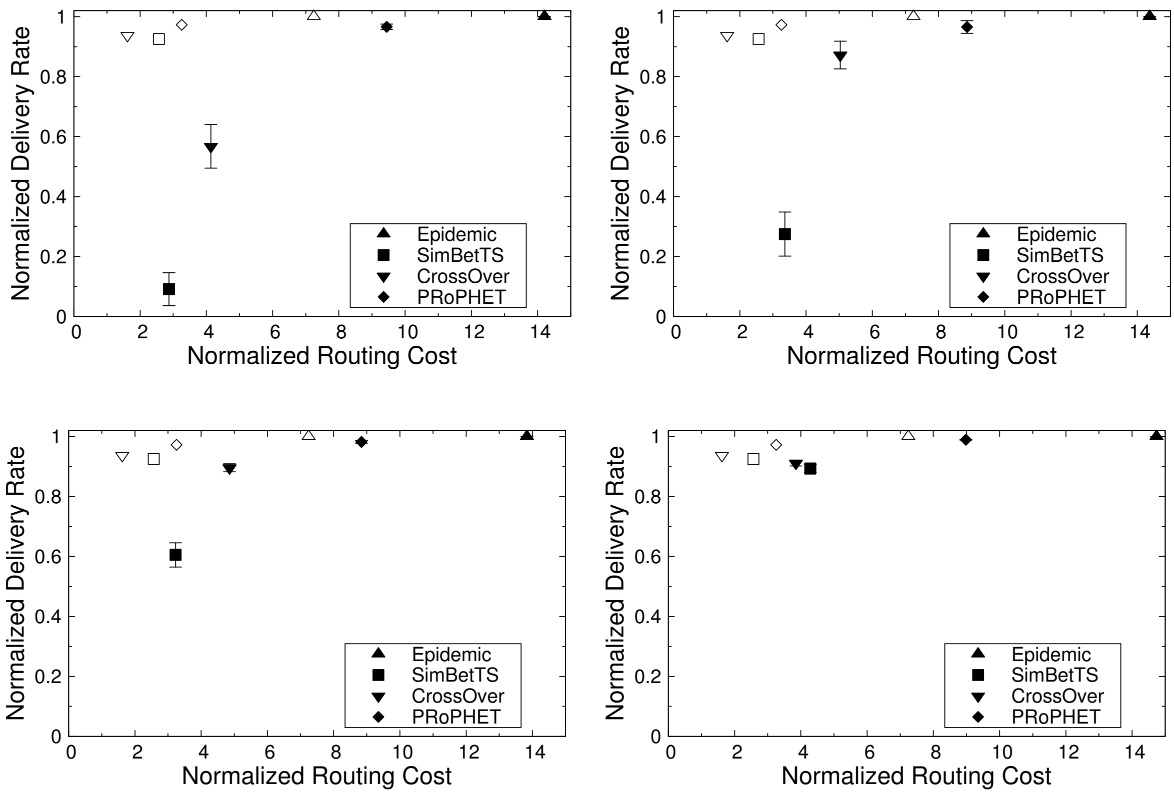


Figure 5.6: C4 scenario - Performance in terms of delivery efficiency and routing cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$

Chapter 6

Conclusions

After a comprehensive study on a variety of real human traces, we showed that they exhibit a very small effective diameter when considering the minimum hop paths among all nodes. Additionally, the effective diameter of these datasets quickly converges to its minimum value regardless of the time we start monitoring the network. We observe that this diameter is likely to increase in large scale networks.

Therefore, we proposed CrossWorld a new paradigm for constructing large scale synthetic traces for social mobile networks. It does not rely in formulating a human mobility model, but uses real world traces as Building blocks and adds links among them. CrossWorld is able to produce synthetic traces that feature a set of realistic attributes, like contact distribution and high clustering coefficient, and an increased degree of separation. A more selective choice of copying contacts is left as future work .

Afterwards, we evaluated state-of-the-art routing algorithms in such environment and showed that they face significant performance challenges in cases with increased degree of separation.

Finally, we proposed CrossOver, a new routing algorithm designed to achieve good performance regardless the size of the network. CrossOver makes routing decisions by using more than one utility metrics, regarding to nodes distance from destination. We evaluate CrossOver and compare it with the performance of state-of-the-art routing algorithms. We showed that it achieves better performance at CrossWorld traces as well as in real world datasets, in terms of delivery ration and transmissions.

Bibliography

[1] Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. Gephi: An open source software for exploring and manipulating networks, 2009.

[2] S. Batabyal and P. Bhaumik. Mobility models, traces and impact of mobility on opportunistic routing algorithms: A survey. *Communications Surveys Tutorials, IEEE*, PP(99):1–1, 2015.

[3] Suzan Bayhan, Esa Hyytiä, Jussi Kangasharju, and Jörg Ott. Analysis of hop limit in opportunistic networks by static and time-aggregated graphs. In *IEEE International Conference on Communications (ICC'15)*, London, UK, Jun. 2015. to appear.

[4] Augustin Chaintreau, Abderrahmen Mtibaa, Laurent Massoulie, and Christophe Diot. The diameter of opportunistic mobile networks. In *Proceedings of the 2007 ACM CoNEXT Conference*, CoNEXT '07, pages 12:1–12:12, New York, NY, USA, 2007. ACM.

[5] Elizabeth M Daly and Mads Haahr. Social network analysis for information flow in disconnected delay-tolerant MANETs. *IEEE Trans. Mobile Comput.*, 8(5):606–621, 2009.

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

[7] Nathan Eagle and Alex (Sandy) Pentland. CRAWDAD data set mit/reality (v. 2005-07-01). Downloaded from http://crawdad.org/mit/reality/, July 2005.

[8] Vijay Erramilli, Augustin Chaintreau, Mark Crovella, and Christophe Diot. Diversity of forwarding paths in pocket switched networks. In *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, IMC '07, pages 161–174, New York, NY, USA, 2007. ACM.

[9] Vijay Erramilli, Mark Crovella, Augustin Chaintreau, and Christophe Diot. Delegation forwarding. In *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '08, pages 251–260, New York, NY, USA, 2008. ACM.

[10] Samo Grasic, Elwyn Davies, Anders Lindgren, and Avri Doria. The evolution of a dtn routing protocol - prophetv2. In *Proc. Workshop Challenged Netw.*, pages 27–30. ACM, 2011.

[11] T. Hossmann, Thrasyvoulos Spyropoulos, and F. Legendre. Know thy neighbor: Towards optimal mapping of contacts to social graphs for dtn routing. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9, March 2010.

[12] Pan Hui, Jon Crowcroft, and Eiko Yoneki. Bubble rap: Social-based forwarding in delay-tolerant networks. *IEEE Trans. Mobile Comput.*, 10(11):1576–1589, 2011.

[13] Wei jen Hsu, Thrasyvoulos Spyropoulos, K. Psounis, and A. Helmy. Modeling spatial and temporal dependencies of user mobility in wireless mobile networks. *Networking, IEEE/ACM Transactions on*, 17(5):1564–1577, Oct 2009.

[14] T. Karagiannis, J.-Y. Le Boudec, and M. Vojnovic. Power law and exponential decay of intercontact times between mobile devices. *Mobile Computing, IEEE Transactions on*, 9(10):1377–1390, Oct 2010.

[15] S. Kosta, A. Mei, and J. Stefa. Large-scale synthetic social mobile networks with swim. *Mobile Computing, IEEE Transactions on*, 13(1):116–129, Jan 2014.

[16] Jérémie Leguay, Anders Lindgren, and Timur Friedman. CRAWDAD data set upmc/content (v. 2006-11-17). Downloaded from http://crawdad.org/upmc/content/, November 2006.

[17] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graphs over time: Densification laws, shrinking diameters and possible explanations. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, KDD '05, pages 177–187, New York, NY, USA, 2005. ACM.

[18] Anders Lindgren, Avri Doria, and Olov Schelén. Probabilistic routing in intermittently connected networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7(3):19–20, July 2003.

[19] Paolo Meroni, Sabrina Gaito, Elena Pagani, and Gian Paolo Rossi. CRAWDAD data set unimi/pmtr (v. 2008-12-01). Downloaded from http://crawdad.org/unimi/pmtr/, December 2008.

[20] Anh-Dung Nguyen, Patrick Senac, and Michel Diaz. Understanding and modeling the small-world phenomenon in dynamic networks. In *Proceedings of the 15th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM '12, pages 377–384, New York, NY, USA, 2012. ACM.

[21] N. Papanikos and E. Papapetrou. Coordinating replication decisions in multi-copy routing for opportunistic networks. In *Wireless and Mobile Computing, Networking and Communications (WiMob), 2014 IEEE 10th International Conference on*, pages 8–13, Oct 2014.

[22] Anna-Kaisa Pietilainen. CRAWDAD data set thlab/sigcomm2009 (v. 2012-07-15). Downloaded from http://crawdad.org/thlab/sigcomm2009/, July 2012.

[23] Anatol Rapoport. Spread of information through a population with socio-structural bias: I. assumption of transitivity. *The bulletin of mathematical biophysics*, 15(4):523–533, 1953.

[24] N. Sastry, D. Manjunath, K. Sollins, and J. Crowcroft. Data delivery properties of human contact networks. *Mobile Computing, IEEE Transactions on*, 10(6):868–880, June 2011.

[25] James Scott, Richard Gass, Jon Crowcroft, Pan Hui, Christophe Diot, and Augustin Chaintreau. CRAWDAD data set cambridge/haggle (v. 2006-01-31). Downloaded from http://crawdad.org/cambridge/haggle/, January 2006.

[26] Tara Small and Zygmunt J Haas. Resource and performance tradeoffs in delay-tolerant wireless networks. In *Proc. Workshop Delay Tolerant Netw.*, pages 260–267. ACM, 2005.

[27] Thrasyvoulos Spyropoulos, K. Psounis, and C.S. Raghavendra. Efficient routing in intermittently connected mobile networks: The multiple-copy case. *IEEE Trans. Netw.*, 16(1):77–90, 2008.

[28] Thrasyvoulos Spyropoulos, T. Turletti, and K. Obraczka. Routing in delay-tolerant networks comprising heterogeneous node populations. *Mobile Computing, IEEE Transactions on*, 8(8):1132–1147, Aug 2009.

[29] Gautam S. Thakur, Ahmed Helmy, and Wei-Jen Hsu. Similarity analysis and modeling in mobile societies: The missing link. In *Proceedings of the 5th ACM Workshop on Challenged Networks*, CHANTS '10, pages 13–20, New York, NY, USA, 2010. ACM.

[30] Amin Vahdat, David Becker, et al. Epidemic routing for partially connected ad hoc networks. Technical report, Duke University, CS-200006,2000.

[31] John Whitbeck, Marcelo Dias de Amorim, Vania Conan, and Jean-Loup Guillaume. Temporal reachability graphs. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, Mobicom '12, pages 377–388, New York, NY, USA, 2012. ACM.

Appendix

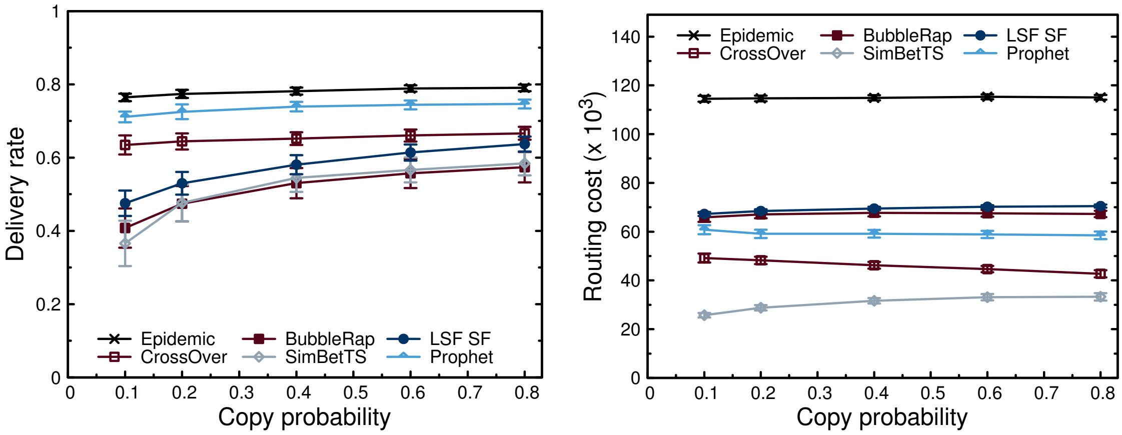


Figure 6.1: RRR scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 1$

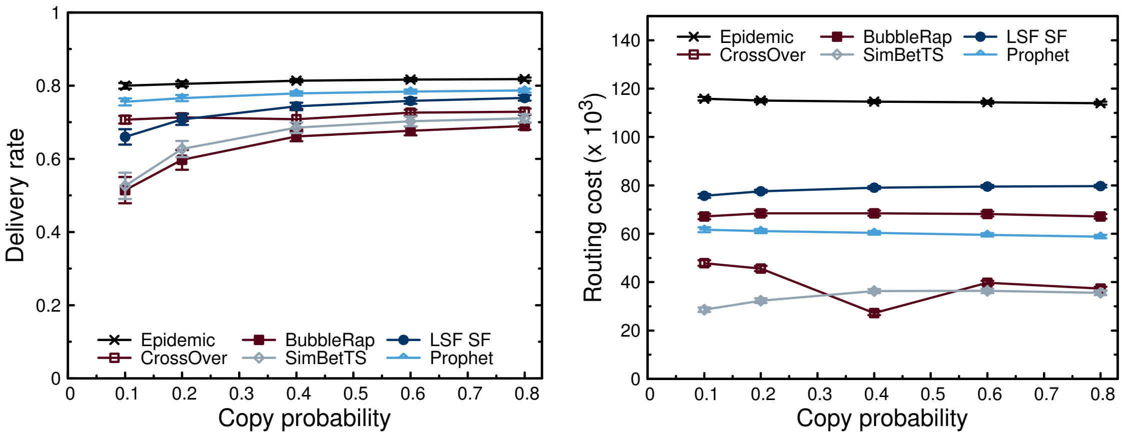


Figure 6.2: RRR scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 2$

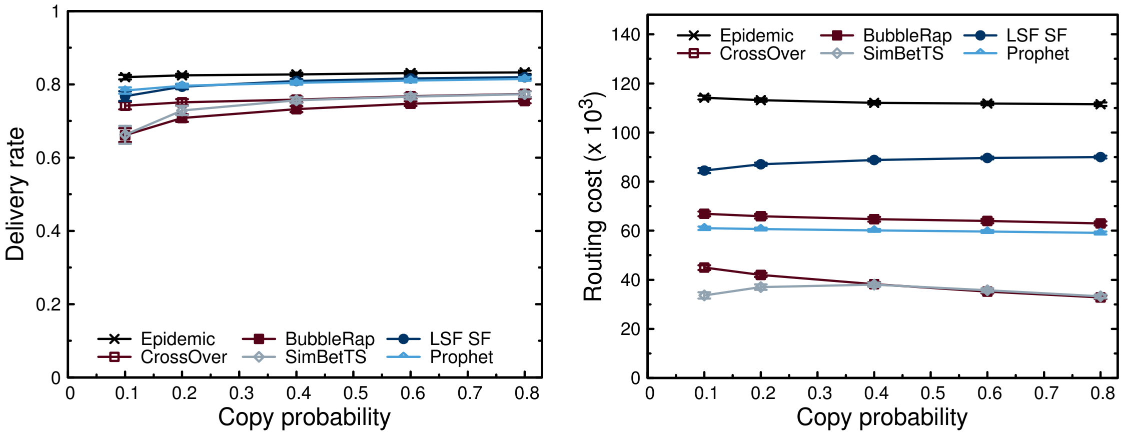


Figure 6.3: RRR scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 4$

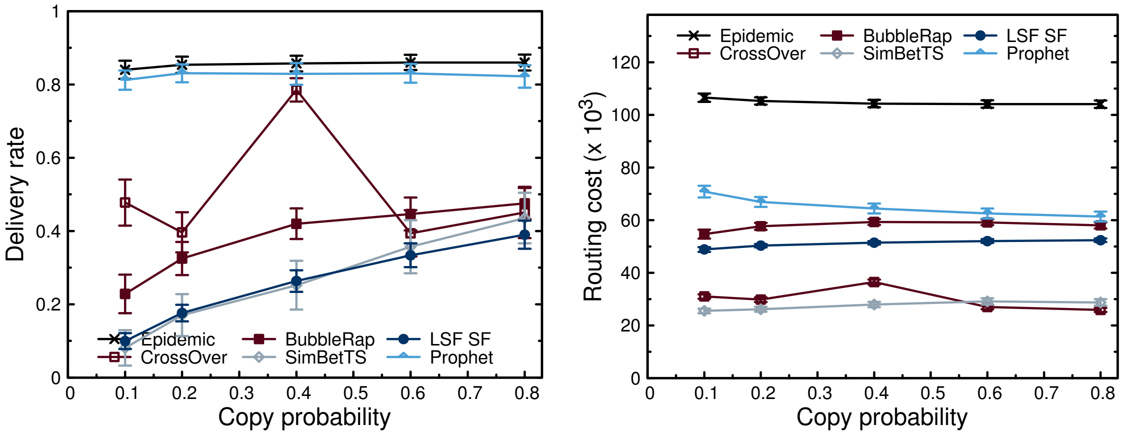


Figure 6.4: C4 scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 1$

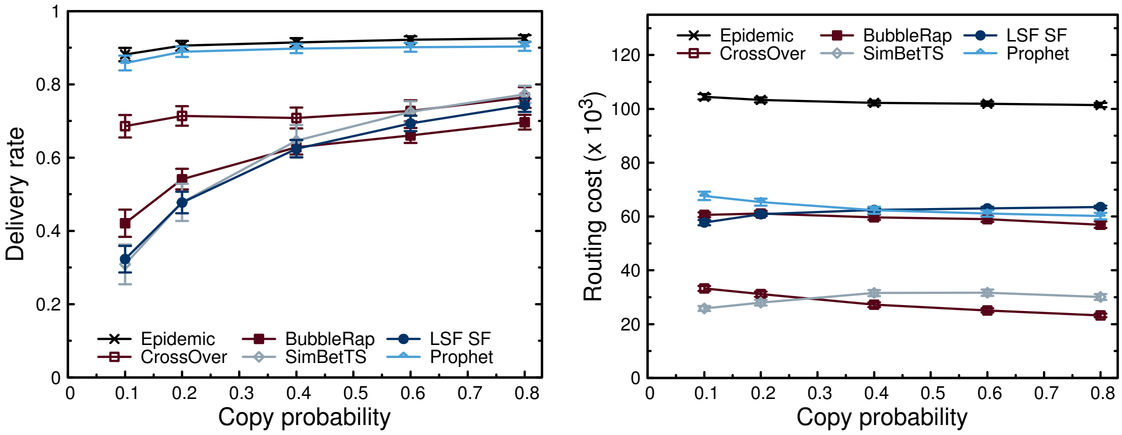


Figure 6.5: C4 scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 2$

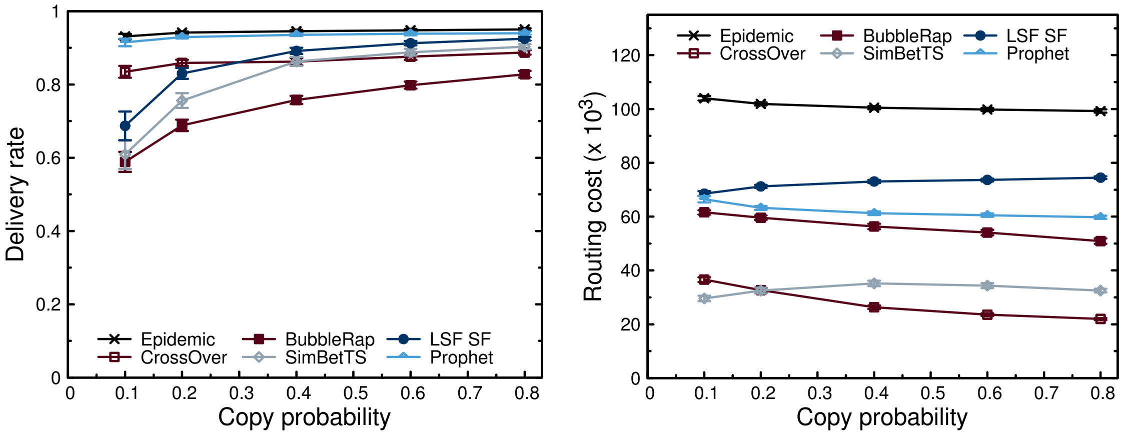


Figure 6.6: C4 scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 4$

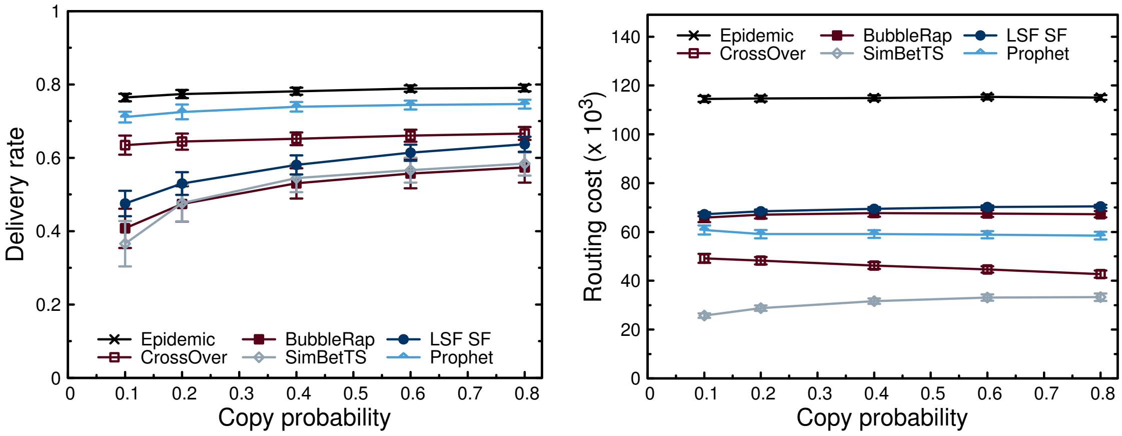


Figure 6.7: RMC scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 1$

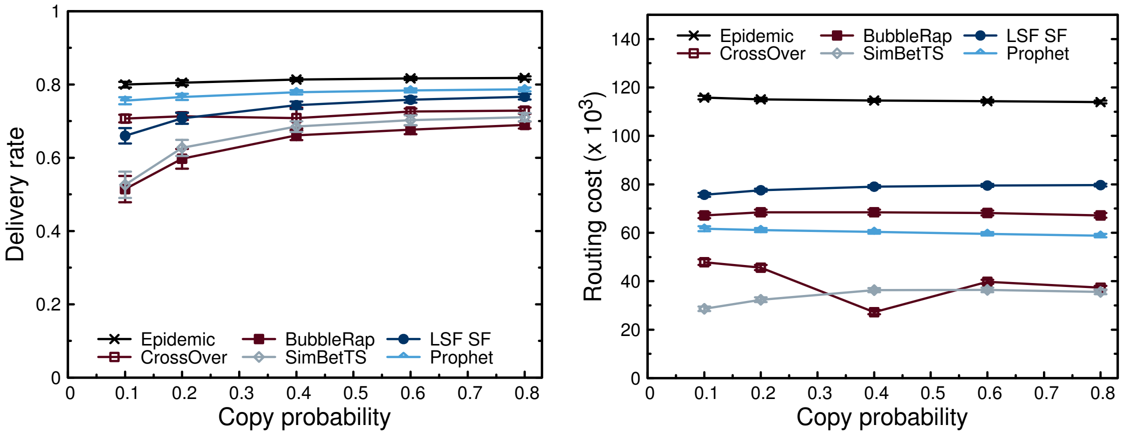


Figure 6.8: RMC scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 2$

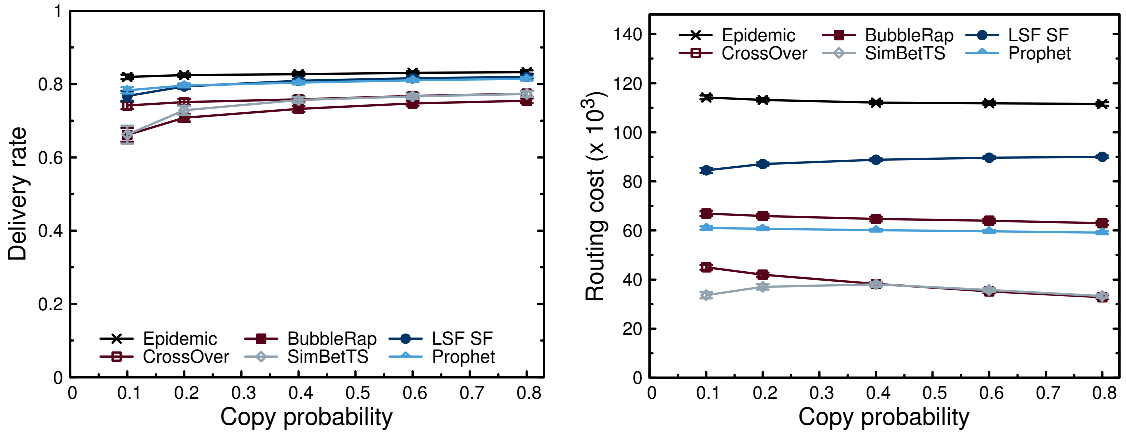


Figure 6.9: RMC scenario - Performance in terms of delivery efficiency(a) and routing cost (b) for $bp = 4$

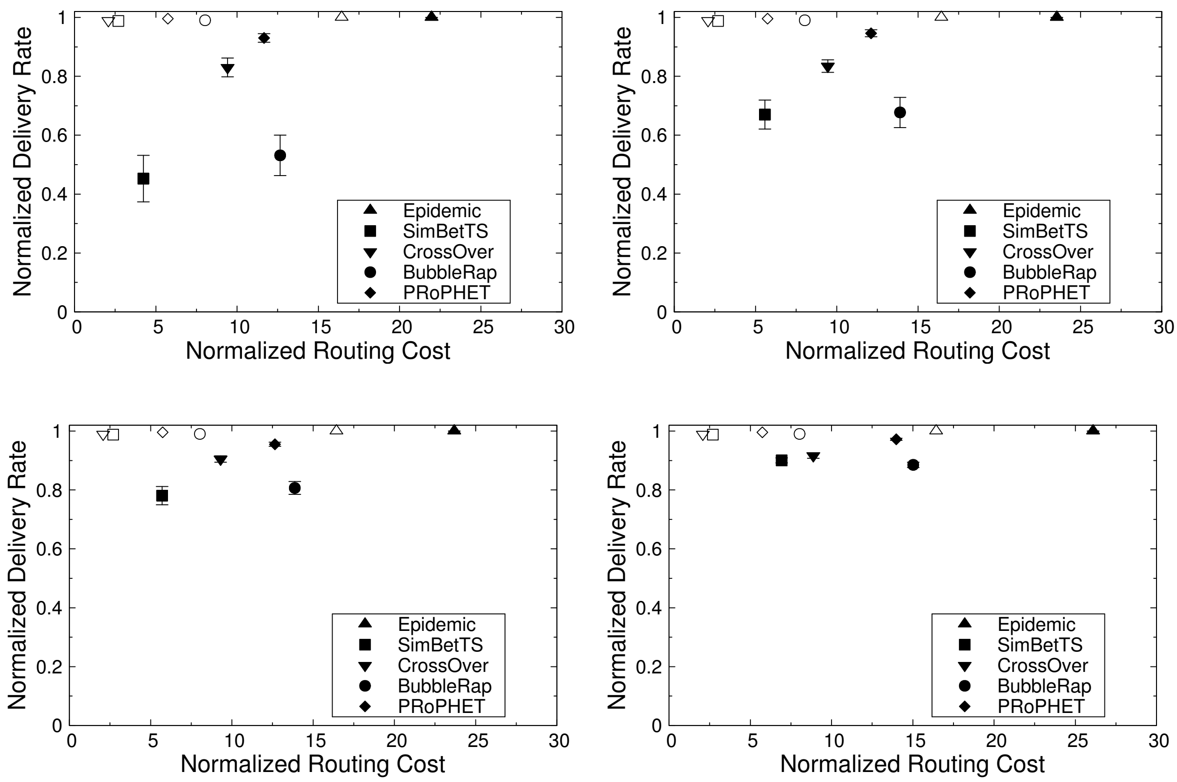


Figure 6.10: C4 scenario - Performance in terms of delivery efficiency and routing cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$

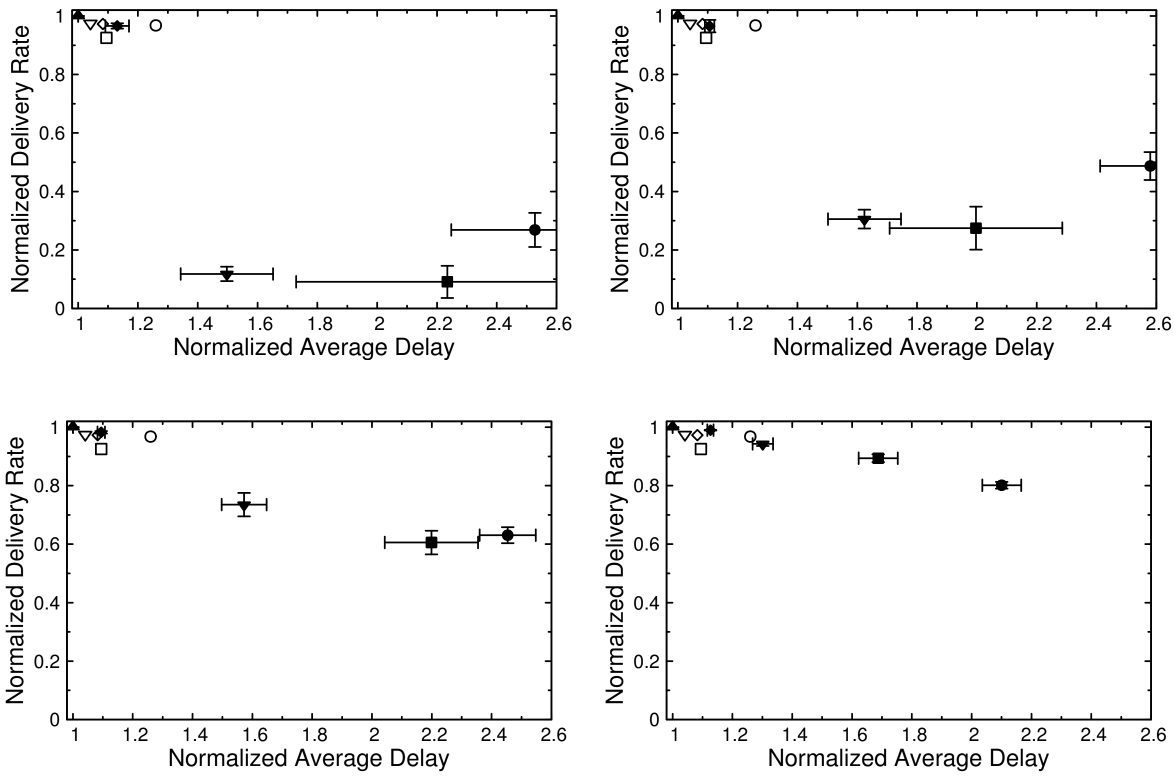


Figure 6.11: C4 scenario - Performance in terms of delivery efficiency and delivery delay cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$

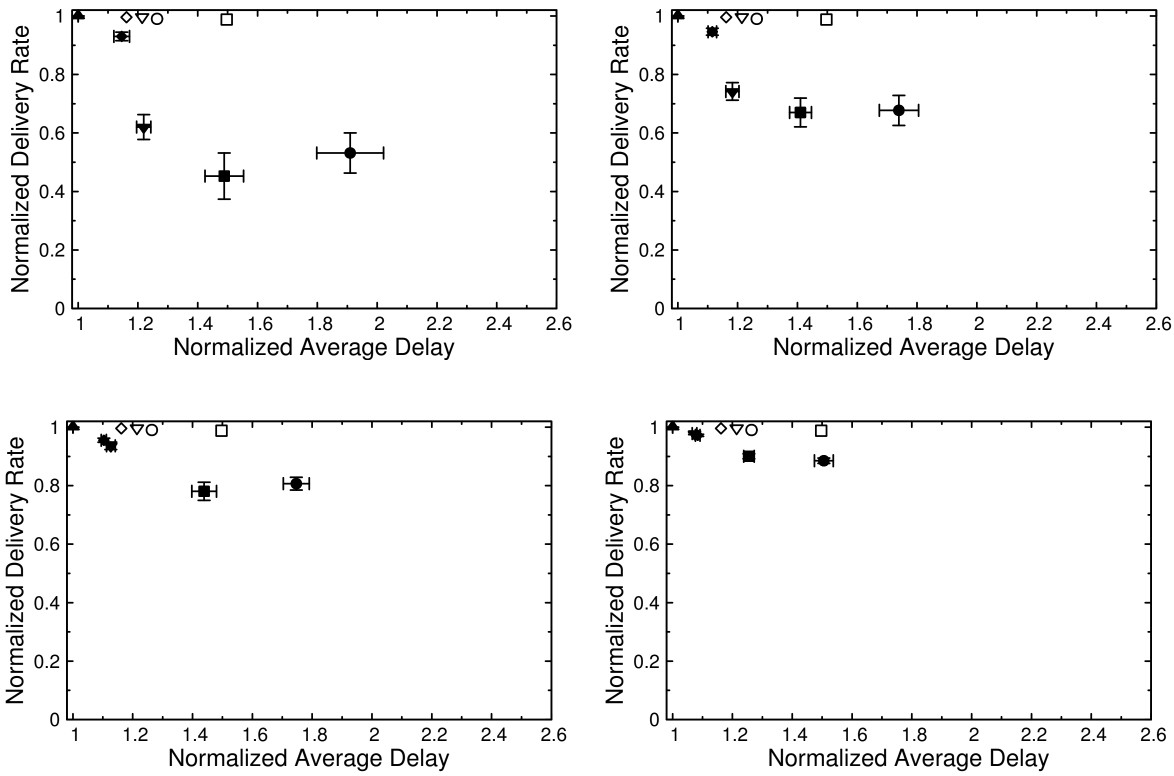


Figure 6.12: C4 scenario - Performance in terms of delivery efficiency and delivery delay cost (a) $p = 0.1$ $bp = 1$ (b) $p = 0.4$ $bp = 1$ (c) $p = 0.1$ $bp = 4$ (d) $p = 0.4$ $bp = 4$

Short Vita

Thanasis Koufoulis was born in Arta, Greece in 1988. He was admitted at the Computer Science Department of Athens University of Economic & Business in 2005. He received his BSc degree in Computer Science in 2011 and he is currently a postgraduate student Computer Science & Engineering Department, University of Ioannina. He is a member of the Network Research Group of the University of Ioannina since 2014. His main research interests lie in the field of Mobile and Opportunistic Networks.